Rimfire 3880 SCSI Host Bus Adapter. . . . .

User's Guide

Ciprico Inc.

Publication No. 21020285 D

Preface

This manual is intended to instruct you on how to install and use the Rimfire® 3880 (RF3880) SCSI Host Bus Adapter. This adapter is essentially compatible with the Ciprico® RF3500 series of adapters.

The Target Mode feature available on the RF3880 allows you to use this adapter as either an initiator or target on the SCSI bus.

This manual provides the information necessary to change an existing driver written for the RF3500 series of adapters, or write a new board driver to take advantage of the advanced operation possible with the RF3880.

Organization

Two levels of information are provided in this manual:

- Reference information
- Practical Usage information

Chapters 3, 5, 6, and 7, together with Appendices A and B, provide specific information concerning installation, hardware jumpers, commands, and error codes. They are meant to be reference sources.

Other chapters — 2, 4, 8, and parts of 3, 5, and 6 —provide examples that explain processes or features of the RF3880. You should turn to these chapters for insights into usage of the RF3880 or a particular feature.

A brief description of each section of the manual follows:

Overview of the RF3880 User’s Guide

Chapter 1: Product Overview - Provides an overview of the RF3880, to acquaint you with the product as a whole.

Chapter 2: Hardware Essentials - Explains, in text and illustration, the basic structure of the RF3880 hardware; port addresses and usage.

Chapter 3: Hardware Installation - Takes a practical approach to physically installing the RF3880 in a VMEbus system and attaching SCSI peripherals.

Chapter 4: Command Operation - Explains the command structures and how to issue a command.

Chapter 5: Initiator Mode Pass-through Commands - Describes the Parameter Block structure used for Pass-through commands in Initiator Mode. These are commands that manipulate a device on the SCSI bus.

Chapter 6: Target Mode Pass-through Commands - Describes the Parameter Block structure used for Pass-through commands in Target Mode, and the additional structure used to communicate to the host when the adapter is operating in Target Mode.

Chapter 7: Board-control Commands - Provides a look-up reference for Board-control commands -- commands that are interpreted by the adapter to affect its operation. These commands are used to tailor board operation for specific systems and applications.

Chapter 8: Details of Usage - Explains in detail how and why to use some of the features of the RF3880 adapter.

Appendix A: Error Codes - Lists the error codes returned in the Error field of the Status Block and also the codes returned in the Status Port.

Appendix B: Cables and Connectors - Provides connector pin-outs and specifies the maximum allowable cable length.

Appendix C: Specifications - Describes the physical and electrical characteristics of the RF3880 adapter.

Appendix D: Defaults - Provides default values used by the RF3880: jumpers, hardware ports and command defaults are presented.

Appendix E: Design Differences - Explains, with page references to the appropriate section of this manual, the differences between writing a driver for the RF3570 and the RF3880.

Conventions

This manual incorporates the following conventions:

Notational

1. An upper case letter H following a number indicates the number is a hexadecimal value (e.g., 32 = 20H and 255 = FFH).

2. In this document, a byte is defined as an 8-bit quantity, a word as a 16-bit quantity, and a double word as a 32-bit quantity.

3. This manual represents all memory ordering in the Motorola format. This means that the least significant byte of a double word is placed in the most significant memory location. For a further discussion of memory ordering and information to use Intel ordering, see Chapter 8.

4. The title of this manual, and all references to the adapter throughout, express the name of the product as Rimfire 3880. This name is a base model number representing a series of product offerings with differences that are noted in the text of this manual when pertinent.

Graphic

To explain the RF3880 command structures and describe the breakdown of the command structure into fields, (double-word, word, and byte wide) this manual represents the command structures and fields graphically, as they would be placed in memory.

Throughout this User's Guide you will find the command structures graphically represented as shown below.

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Double-word			
04H	Byte	Byte	Word	
08H				
0CH				
10H				
14H				
18H				

Table 1: Graphic Representation of Parameter Block

The left hand column and top two rows of the above table provide a reference for the position of the components of the structure in memory.

Address Offset - The grey-shaded area on the left is named Address Offset. Since the structures are double-word wide, these hexadecimal values, added to the Starting Memory Address, reference the beginning of each double word in the structure.

Byte Memory Address - The area at the top of the graphic is the Byte Memory Address. This displays the ordering of byte-wide fields of memory. Each of the four "Offset + x" fields describes the relationship of byte fields to the Address Offset.

Reference Documents

The following documents apply to the Ciprico Rimfire 3880 SCSI host bus adapter and may be of interest to you:

VMEbus Specification Manual, Revision D (Motorola Publication)

American National Standard: Small Computer System Interface (SCSI-1) Standard, (ANSI X3.131, 1986)

American National Standard: Common Command Set, ANSI X3 Working Committee Document, X3t9.2/85-52 Rev.4A

Draft Proposed American National Standard: Small Computer System Interface-2, (ANSI X3.131, 199X)

Draft Proposed American National Standard for information systems-SCSI-3 Parallel Interface

User manuals for applicable SCSI devices.

Table of Contents

1

Product Overview

This chapter provides an overview of the RF3880 product and is intended to orient you to the product as a whole, before focusing on specific areas in the chapters that follow.

In this chapter you will find information about:

- Features of this design
- Where to find additional information

This chapter is most valuable to someone who wishes to find out general information about the RF3880 product.

About the Rimfire 3880 Design

The Rimfire 3880 contributes to the family of VME/SCSI adapters already available from Ciprico with a unique design that addresses the particular needs of high bandwidth applications (applications that require large amounts of data to be transferred quickly in order to be interpreted for use—applications such as imaging or data acquisition for weather prediction, for example). These applications are different from transaction intensive applications that focus on low command overhead. High bandwidth applications require:

- extremely fast data transfer rates
- highly efficient use of native bus structures

With its buffering scheme and its effective co-ordination of the VMEbus and SCSI bus, the RF3880 fulfills both requirements for high bandwidth applications.

Latest Bus Specifications

Part of the RF3880's contribution to the needs of high bandwidth applications is derived from designing to the latest bus specifications for VME and SCSI. To make the most of the VMEbus capabilities, the RF3880 implements 64-bit-wide block transfer mode. On the SCSI side, the QLogic FAS256 chip is effectively applied to implement Fast, 16-bit SCSI data transfers.

Buffering scheme

Additionally, the RF3880 design maximizes the efficiency of both buses by using a data buffer to decouple one bus operation from the other.

- Data transfers on the SCSI bus can occur completely independently of the VMEbus activity.
- Data on either side of the buffer can flow in opposite directions
- The buffer is large enough (512 Kbytes) to be used for temporary storage.

Startup delays on the SCSI bus are eliminated because data (or space) in the buffer is available as soon as the SCSI device switches to data phase, thus increasing system throughput. On the VME side, command I/O processing can occur at the same time data is being transferred to/from the SCSI bus.

Buffer Management

It is in the buffer management technique employed by the RF3880 that the unique dedication to bandwidth applications is realized. The RF3880 hardware logic allows the buffer to be managed either as a large FIFO or as temporary storage of data from several commands.

The adapter dynamically adjusts the management of the buffer based on the type and size of the data transfers involved.

- For large block transfers (greater than 4 Kbytes) the adapter will setup the buffer like a FIFO. Both the SCSI and VME sides of the buffer are reserved for the transfers. In this mode, the hardware performs and manages the transfer, then signals completion. The advantage to this algorithm is that microprocessor involvement is minimal.
- On transfers of small data blocks, (less than or equal to 4 Kbytes) the adapter handles the buffer more like a cache. In this mode, each side of the buffer can act independently; data transfers can occur simultaneously, in response to separate commands. The advantage of this algorithm is that data is already in the buffer when it is ready to be processed—SCSI overhead is overlapped with data transfer on the VME side.

Target Mode Implementation

The RF3880 is unique in its implementation of Target Mode. When Target Mode is enabled, the adapter can act as an initiator and target on the SCSI bus. When operating as a target, the adapter passes SCSI command information to the Host. It is up to the Host to instruct the adapter what action to take. This provides an enormous flexibility for implementation.

What next?

Depending on what you wish to do next, you should turn to one of the following chapters:

To learn more about the RF3880 hardware:
— Continue to Chapter 2.

To learn more about the Installation process:
— Go to Chapter 3.

To learn more about the Command Operation:
— Go to Chapter 4.

To learn more about Initiator Mode Pass-through commands:
— Go to Chapter 5.

To learn more about Target Mode Pass-through commands:
— Go to Chapter 6.

To learn more about the Board-control commands:
— Go to Chapter 7.

To learn more about usage:
— Go to Chapter 8.

2

Hardware Essentials

Introduction

This chapter explains, in text and illustrations, the basic structure of the RF3880 hardware and is provided to assist you with installing and communicating with the board.

Summary

In this chapter you will learn more about:

- Structure of the hardware based on the VLSI used.
- Hardware ports used to begin communication to the board.

This chapter is useful for anyone wishing to know what technology was used to design the RF3880 and understand the usage of the product.

Hardware Structure and Description

The RF3880 hardware design incorporates the technology available from the following components:

- Intel 20 MHz 80186XL microprocessor
- QLogic FAS256 16-bit, Fast SCSI Controller chip
- High Density Programmable Logic Devices (PLD's) for VMEbus, and buffer control
- Ciprico Pipelined System Interface (PSI)

Intel 80186XL

At the core of the RF3880 design is the 80186XL supervisory microprocessor. The 80186 is well suited to the task of overseeing board operations; it uses optimized instruction encoding for high performance and memory efficiency. RF3880 performance is further boosted by using the 20 MHz version of the 80186.

QLogic FAS 256

SCSI bus protocol for the RF3880 is efficiently handled by the QLogic Fast SCSI chip. The QLogic chip provides the capability for 16-bit transfers on the SCSI bus, as well as negotiation for Fast Synchronous transfers across the bus, up to 20 MB/s. The QLogic FAS256 chip is a registered device that can be set-up for automated bus operations. These options are available to you via the Board-control commands discussed in Chapter 7.

Pipelined System Interface

Ciprico created the custom VLSI Pipelined System Interface (PSI) to coordinate access to the system address bus; it has registers for the DMA address count and transfer count. Also contained in the chip are the Channel Attention port register and the Interrupt Request level register.

Data Buffer

Using high-speed video DRAMs to provide 512 Kbytes of buffer space with hardware control logic, the RF3880 creates a data buffer that adjusts dynamically to the type and size of the data transfers involved. Performed in hardware, this design maximizes performance by bypassing the need for microprocessor involvement.

Block Diagram

The figure below is a block diagram of the RF3880; it illustrates the hardware just described. The High Speed VMEbus DMA and The VMEbus Interrupt Generator blocks are part of the PSI.

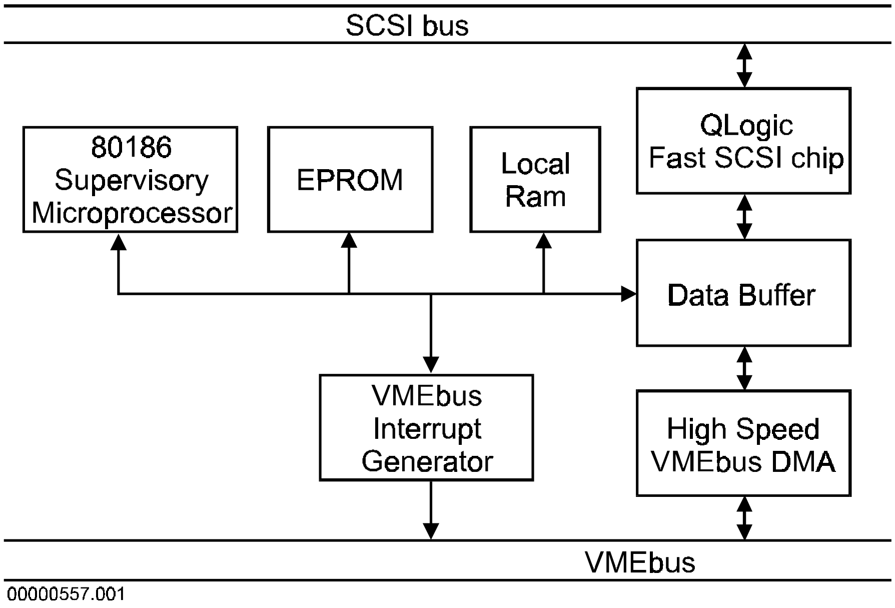


Figure 1 Block Diagram of RF3880

Board Addressing

The Board Address and Slave Address Modifier to which the RF3880 responds on the VMEbus are set with jumpers on the board.

Choosing an Address Modifier

The RF3880 adapter is an A16 SLAVE device on the VMEbus. In keeping with VMEbus specifications, the Address Modifier for selection of this type of device can be either 29H or 2DH. Since there is only one bit difference between the two values, a single jumper is provided to select an Address Modifier.

AM5	AM4	AM3	AM2	AM1	AM0
1	0	1	Jumper In = 0 Out = 1	0	1
VMEAddress Modifier Lines					

Table 2: Choosing an Address Modifier

Choose Address Modifier 2DH by removing the jumper, when you wish to reserve the RF3880 registers for short supervisory access. Choose 29H by inserting the jumper, if you wish to allow short non-privileged access:

Hex Code	Address Modifier						Function
	5	4	3	2	1	0	
2D	1	0	1	1	0	1	Short Supervisory Access
29							Short Non-Privileged Access

Table 3: VMEbus Address Modifiers for RF3880

The factory setting for the RF3880 Slave Address Modifier is 2DH; the jumper is out.

The location and setting of this jumper is described in Chapter 3, Hardware Installation.

Choosing a VMEbus Address

With the address jumpers on the RF3880 board, you set the upper 7 bits of the Board Address. This effectively reserves 512 bytes of VMEbus 16- bit address space. Within the reserved area, the RF3880 adapter has four register ports that are the foundation of its communication scheme.

A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
Address lines decoded to identify the RF3880 base address which is set via jumpers on the board.											Address lines decoded to select one of four ports.				

Table 4: Address Line Decoding

The factory setting, for the RF3880 VMEbus Address, is to jumper A12, making the default base address EE00H. Details about the location and setting of the jumpers can be found in Chapter 3, Hardware Installation.

Hardware Ports: General

There are four ports that are used to communicate with the RF3880 adapter: Address Buffer Port; Channel Attention Port; Status Port; and Reset Port. The address and function of each port is described in this section.

Hardware Port Addresses

Each of the Hardware ports can be addressed as an offset from the base address, (which is set with jumpers on the board). Further, all ports can be accessed in 16-bit mode. To address a specific port you must add the correct offset to the base address value. The offset values for the RF3880 hardware ports are described in the figure below.

Port	Address Offset	Host Read or Write
Address Buffer	00H	Write
Channel Attention	08H	Write
Status	10H	Read
Reset	18H	Write

Table 5: Hardware Port Addresses

Example Port Addresses - Using the factory set Base Address for the RF3880, EE00H, the Hardware Port Addresses would be:

Address Port = EE00H
Channel Attention Port = EE08H
Status Port = EE10H
Reset Port = EE18H

The table above also indicates how each port can be accessed by the Host: read or write.

Overview of the Ports

An overview of the operation of all of the Hardware Ports is provided next. The overview is followed by a detailed description of each Hardware Port that focuses on the rules of operation.

Address Buffer Port - This port is instrumental in setting up the RF3880 adapter I/O operation. With it you can do two things:

1. Set system bus transfer characteristics such as bus width and byte/word swapping for subsequent operations. Usually this is done only with the first command sent to the board, when you are initializing the board for operation in your system.

2. Pass the address of a Single Command Structure to the adapter. Any time you wish to issue a Single Command, you must use the Address Buffer Port to pass the address structure. Most often, the first Single Command executed contains a command to Start Command List. This function of the port is only used when issuing a Single Command.

Channel Attention Port - The Channel Attention Port is your means of informing the RF3880 adapter that a command is ready to be executed. The value you write to this port tells the adapter what kind of Command Structure to expect. Before you write to the Channel Attention Port you must set-up the appropriate Command Structure in the system memory space you will be directing the adapter to read.

Status Port - This port is used to read status from the adapter. There are two types of status that can be read: General Status—returned after a command completes during normal operation; and Reset Status—returned during board self-test at reset or power-up.

Reset Port - With the Reset Port you are able to initiate, through software, a reset identical to a hardware reset.

Address Buffer Port

The Address Buffer Port is a write-only port. It is used when you wish to execute a command using a Single Command Structure (SCS). It is loaded with: the starting system address of the structure; an

address modifier value to be used when reading the Parameter Block contained in the structure; and a byte of control information through which you may instruct the RF3880 adapter about subsequent operation.

The Address Buffer Port register is three words deep and 16 bits wide. To pass all three words requires three writes to the port:

- The first write must contain the Control byte and Address Modifier.
- The second write contains the Most Significant Word (MSW) of the Single Command Structure address in system memory.
- The last write contains the Least Significant Word (LSW) of the Single Command Structure address.

The order of these writes is illustrated in the following figure:

Bits	15	8	7	0
1st Write	Control		AM Bits for SCS	
2nd Write	Single Command Structure Address: MSW			
3rd Write	Single Command Structure Address: LSW			

Table 6: Address Buffer Port Writes

A description of each of the fields of the Address Buffer Port follows.

Address Modifier Field

The Address Modifier (AM) Field of the Address Buffer Port is used to specify the Address Modifier value that the RF3880 adapter should use when reading the first Parameter Block.

Control Field

Use this byte-wide field to tailor the RF3880 adapter’s use of the system bus. With it, you can set-up the adapter to:

- Transfer data in 16-bit, 32-bit or 64-bit mode on the VMEbus.
- Byte or word swap command structures to be compatible with the byte and word ordering of either Intel or Motorola architectures.

Byte and word swapping are further discussed in Chapter 8, Details of Usage.

The bit meanings of the Control Field are as follows:

7	6	5	4	3	2	1	0
SET	0	0	0	0	WID	WSC	BSC

Table 7: Control Field of Address Buffer Port

Note
For any of the control bits to be considered valid by the adapter, the SET bit must be one.

BSC Byte Swapping Control - Defines whether high and low bytes of a word will be swapped during transfer of status and parameter blocks.

0 ⇒ No swapping.
1 ⇒ Change byte order.

WSC Word Swapping Control - Defines whether high and low words of a double word will be swapped during transfer of status and parameter blocks.

0 ⇒ No swapping.
1 ⇒ Change word order.

WID WIDth of data transfers - Determines (in conjunction with VMEbus Address Modifier) the width of data transfers across the system bus.

Note
The RF3880 supports 64-bit wide block transfers for only those Address Modifiers listed in the VME specification.

0 ⇒16-bit data transfers.
(forces 16-bit transfers regardless of Address Modifier used.)
1 ⇒32-bit or 64-bit data transfers, dependent on the Address Modifier used.

SET select new control SETtings - Determines whether the rest of the bits in the Control Field should be applied to adapter operation.

0 ⇒ ignore all other bits.
1 ⇒ apply bit values to next transfer.

Determining a Control Field value - These are the significant points to remember when determining the value you should use for the Control Field:

- The Control Field value is meant to be set once, usually during system initialization. Generally, its value will be set with the first command issued to the board.
- The values you choose for the Control Field will be applied immediately. They will be applicable for the next parameter block read in and all future system memory accesses.
- The Control Field value is never applied to the Address Buffer Port or any of the other hardware ports. Always follow the format given in this section when accessing the hardware ports.
- Attempting to change the Control Field setting after the RF3880 adapter has begun executing a command list will cause unexpected results.
- The default value for the Control Field is zero; the adapter will operate in 16-bit mode and the parameter and status block structures will not be swapped.

Single Command Structure Address Fields

This two-word address is the starting address of the Single Command Structure containing the command to be issued to the board.

The Address Buffer Port is only used when issuing a Single Command to the RF3880 adapter.

Channel Attention Port

The Channel Attention Port is a write-only port. It is used to direct the RF3880 adapter to execute Parameter Block(s) associated with a Command Structure you have already built in system memory. The Channel Attention Port is used for both Single and Command List issue. However, it is written slightly differently in each case (as shown below).

Single Command Channel Attention - To initiate execution of a Single Command you must do the following:

1. Construct a Single Command Structure in system memory.
2. Load the Address Buffer Port with the Control Byte, Address Modifier and the address of the Single Command Structure constructed in Step 1.
3. Write a 0 to the Channel Attention Port, instructing the RF3880 adapter to execute the Parameter Block contained in the Single Command Structure.

   Use an offset of 8H from your Base Address when writing to the Channel Attention Port.

Command List Channel Attention - To execute commands via Command List, you must do the following:

1. Have an active Command List. (Command Lists are begun by issuing the Start Command List command via a Single Command Structure.) If you attempt to issue a command via Command List when there is not an active List, you will receive an error code of 11H in the Status Port.

2. Fill the next available Parameter Block in the Command List with the command you wish to have executed.

3. Adjust the Command List Parameter Block IN Index to indicate additional commands have been added.

4. Write a 1 to the Channel Attention Port, instructing the RF3880 adapter to begin executing the Parameter Block(s) contained in the Command List structure.

   If you have multiple Parameter Blocks ready in the space allocated for Command List, you may issue separate Channel Attention writes or you may issue one for the entire group.

   Use an offset of 8H from your Base Address when writing to the Channel Attention Port.

For more information about how to use the Command List Structure, see Chapter 4, Command Operation.

BIT		
2	1	0
0	TYP	

Table 8: Channel Attention

Channel Attention Port Format

The Channel Attention Port is three bits wide with the following meaning:

TYP Channel Attention Type - Indicates the type of command (or acknowledgment) being issued with a write to the Channel Attention Port.

00 ⇒ Single Command Issue.
01 ⇒ Command List Issue.
10 ⇒ Target Mode Selection area has been read.

(See Chapters 6, 7, and 8 for more information about Target Mode.)

Status Port

The Status Port is a read-only port. It contains information regarding the condition of the board during power-up/reset and during normal board operation. It is used to report:

- A sequence of self-test codes while a reset is occurring.
- An error code if self-test fails or a catastrophic error occurs during normal operation.
- Readiness of the board for commands (RDY bit).
- Readiness for a new Single Command (ENT bit).

The Status Port is 16 bits wide. The bit meanings of the port will be different depending on whether the status is General information or Reset information; each has separate formats. Both types of format are described in this section.

Status Port: Reset Format

The RF3880 adapter performs a series of self-tests on reset or power-up. The Status Port is used by the RF3880 adapter to indicate the progress of the self-test as it is occurring. At power-up or reset, the Status Port will have this format:

BIT															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0								Self-test Code (STC)						RDY	STC

Table 9: Status Port: Reset

Upon power-up or reset the following sequence will occur:

1. Soon after reset, the Status Port is written with a value of FDH.
2. Next, self-test begins and as it progresses, the Self-test Code fields are updated with the code of the test currently executing.
3. The RDY bit is zero during self-test to indicate the board cannot accept any command. All other fields of the Port (except Self-test Code (STC) fields) also remain zero as long as self-test continues.
4. The RDY bit will change to one if the self-tests complete successfully. The self-tests takes five seconds. If, after this time, the RDY bit is still zero, a self-test has failed.

   If a self-test fails, the RDY bit will remain zero and the code of the test that failed will remain in the Self-test Code fields of the Status Port.
5. After the RF3880 self-tests complete successfully, and the RDY bit is set to one, the Status Port format will change to the General format.

Note
Whenever the board is reset, the host operations should wait a minimum of five seconds before testing the RDY bit. If, after the delay, the RDY bit is not set to one, the value in the Self-test Code field of the Status Port will indicate the failing test. The RF3880 will not accept commands if this occurs.

The Status Port fields during power-up or reset, have these meanings:

STC Self-test Code - This bit works in conjunction with bits 2 through 7, to indicate the self-test currently executing.

RDY Ready - This bit indicates when the adapter is ready to accept commands after a reset has completed.

0 ⇒ No command can be accepted.
1 ⇒ Ready to accept a command.

Self-test Code - The values of these bits change as self-test progresses. As self-test begins and progresses, the values are updated to the code of the test currently executing.

See below for a table of Self-test Codes.

Status Port Error Codes

If one of the self-tests does not properly complete, its test number will be left in the Self-test Code byte. The following table describes the test numbers:

Code	Description
00H or FDH	The onboard processor is not working; it cannot access firmware EPROM; cannot access the Status Port.
04H	Static RAM error. All words of static RAM are written with FFFFH, then zero. This test is not exhaustive.
0CH	Firmware CRC error. The calculated CRC for odd and even EPROMs does not match the value stored in EPROM.
80H	SCSI PSI Address/Counter registers not masked to WORD boundaries.
84H	SCSI PSI Address/Counter registers no masked to DWORD boundaries.
88H	Value Miscompare errors of SCSI PSI R/W registers in Manual Load mode.
8CH	Value Miscompare errors of SCSI PSI R/W registers in Auto Load mode.
90H	VME PSI Address/Counter registers not masked to WORD boundaries.
94H	VME PSI Address/Counter registers not masked to DWORD boundaries.
98H	Value Miscompare errors of VME PSI R/W registers in Manual Load mode.
9CH	Value Miscompare errors of VME PSI R/W registers in Auto Load mode.
A8H	SCSI termination power fuse is blown, but termination power is still present at the SCSI connector.
ACH	SCSI termination power fuse is blown and termination power is not available at the SCSI connector.
B0H	Value Miscompare error of QLogic Fast SCSI chips R/W registers.
B4H	QLogic Fast SCSI chip did not generate a reset interrupt.
B8H	QLogic Fast SCSI chip registers not zeroed after reset.
BCH	SCSI bus hung, waiting for reset from QLogic Fast SCSI chip. Could be cabling.
D0H	Video RAM buffer error.
E5H	The onboard processor failed the accumulator test.
E8H	The onboard processor failed the store instruction test.
E9H	The onboard processor failed the shift instruction test.
ECH	The onboard processor failed the compare instruction test.
EDH	The onboard processor failed the jump instruction test.

Table 10: Status Port Error Codes

Status Port: General Format

After the RF3880 self-tests have completed successfully and the RDY bit is set to one, the Status port is used to return general board information. The Status Port will have the following format:

BIT															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Code								0			ERR	0	0	RDY	ENT

Table 11: Status Port: General

The meanings of the Status Port fields are as follows during normal operation:

ENT Entered - This bit is used to signal when a new Single Command may be issued to the adapter via a Channel Attention. The board is ready for another Single Command write to the Channel Attention Port each time this bit value toggles. It does not indicate that a command has completed, only that a new Single Command may be initiated.

Note
This bit does not toggle for Command List Channel Attentions.

Initially this bit ⇒ 0

After 1st Single Command Channel Attention ⇒ 1;

After 2nd Single Command Channel Attention ⇒ 0; etc.

RDY Ready - This bit indicates when the adapter is ready to accept commands after a reset has completed. During normal operation it should always be set to one.

0 ⇒ No command can be accepted.
1 ⇒ Ready to accept a command.

ERR Error - This bit is used to signal that a catastrophic error occurred that could not be reported through the command interface. (Example: Using an invalid Parameter Block address which resulted in a VMEbus error. Typically the AMBER LED will also light if this bit is set.)

0 ⇒ Normal operation.
1 ⇒ Error condition.

Code - Board Type or Error Code - When the ERR bit is set to one, this byte will contain a catastrophic error code. If the ERR bit is zero, the code byte will contain the Ciprico board type.

02 ⇒ Any Ciprico VME/SCSI product.
Other than 02 ⇒ Catastrophic Error Code.

There are several catastrophic error codes:

Code	Name	Description
10H	Bad PBIN or SBOUT Value	The Command List PBIN or SBOUT value exceeded the value set for PBNUM or SBNUM.
11H	State wrong for Start/ Stop Command List command	The code is returned under these conditions: **Start Command List** command issued a command list is already active. **Stop Command List** command issued when no command list is presently active. **Command List Channel Attention** issued when no Command List is active.
12H	Target Mode Error	A **Target Mode Selection Area Read** Channel Attention issued when no selection are was written by adapter.
14H	Software VMEbus Timeout	The adapter times each of its VMEbus transfers; it will report this if a VMEbus transfer takes too long to complete. This can occur for transfers of parameters as well as data.
15H	VMEbus Error Occurred	A VMEbus error was detected while the adapter was transferring either parameters or data.
96H	Internal Firmware Error	This error code indicates that a Firmware error was detected during the execution of a command.

Table 12: Status Port Catastrophic Error Codes

To access the Status Port, use an offset of 10H from the base address of the adapter (which is set by jumpers on the board).

Reset Port

This is a write-only port. You may write to this port when you wish to initiate a board Reset. The write can consist of any value because the data is ignored. The result of this write will be a hardware reset.

Whether or not the RF3880 adapter also resets the SCSI bus at this time is determined by how you set a jumper on the board. (Information about the configuration jumpers is in Chapter 3, Hardware Installation.)

To access the Reset Port use an offset of 18H from the base address of the adapter (set by jumpers on the board).

When a reset has completed with no self-test errors, the RDY bit of the Status Port will be set.

Descriptions of Port Usage

The following describes port usage. Assume that the adapter address jumpers are set to a base address of EE00H.

Issuing a Reset — Write to Reset Port at address EE18H. The RF3880 will begin executing self-test. Read the Status Port at address EE10H to find when self-test completes.

Issuing a Single Command — Write three words to the Address Buffer Port at address EE00H. These words select system bus width/swapping options as well as the Address Modifier and memory address of the Single Command. To begin command execution, issue a Single Command Channel Attention by writing a 0 to Channel Attention Port at address EE08H. Read Status Port at EE10H to check for Enter condition before issuing the next command.

Issuing a Command List — Once an active Command List has been established, write a 1 to the Channel Attention Port (address EE08H) for a Command List Channel Attention after the new commands are added to the list

Acknowledging Target Mode Selection — Write a 2 to the Channel Attention port (address EE08H) to indicate that target data has been read by Host.

Read Board Command Status — Read the Status Port at address EE10H to find the current status of the adapter.

3

Hardware Installation

Introduction

Unifying the parts of a system into a whole requires a smooth transition of the theoretical into the actual. This chapter takes a very practical approach to physically installing an RF3880 board into a VME-based system and attaching SCSI devices.

Summary

In this chapter you will learn:

- The hardware installation process from unpacking to power-up.
- What configuration changes to make.
- What to look for in the documentation that comes with your SCSI device.

This chapter is most helpful for someone who is responsible for making the hardware operational.

Overview of the Installation Process

There are only five steps involved in bringing your RF3880 adapter into functionality. Each is outlined briefly below and then in more detail on the following pages.

Step 1: Unpack the board

The RF3880 adapter is carefully packaged to prevent damage and exposure to static. You should observe precautions while unpacking.

Step 2: Set jumpers and termination on the board

The board has four separate blocks of jumpers. These jumpers are for: default SCSI ID, default SCSI bus reset on power-up, enable SCSI parity-checking, Board VMEbus address and Address Modifier, SYSFAIL signal assertion, and Bus Arbitration.

You will also need to determine whether you will terminate the SCSI bus on the board. You will need to remove termination if the RF3880 adapter is not at the physical end of the SCSI bus cable.

Step 3: Insert the board into a VME slot

You must select a VME slot and prepare it for the adapter by removing Bus Grant and Interrupt Acknowledge daisy-chain jumpers from the VMEbus backplane. There are many types of system enclosures that the RF3880 will fit into.

Step 4: Apply power and observe LEDS

The RF3880 adapter has a set of LEDs on its faceplate that will indicate its state on power-up. You should observe these LEDs and other indicators that are described in this section. If the LEDs indicate a problem, you may need to follow some of the troubleshooting hints also described.

Step 5: Configure and connect SCSI devices

There are several types of SCSI devices that you may be attaching to the RF3880 adapter. Each requires set-up and each must be cabled to the SCSI bus. There are restrictions on the length of the bus cable and termination factors that must be considered.

Step 1: Unpacking the Board

The RF3880 adapter is shipped to you in one box. You should find the following items included in the shipment:

1 RF3880 adapter board
1 Rimfire 3880 SCSI Host Bus Adapter User’s Guide

Shipping Damage

Immediately after receiving it you should check your shipment for evidence of damage or mishandling in transit.

Ciprico’s warranty does not cover shipping damages. Therefore, if the shipping carton is water stained or otherwise damaged, contact the carrier or shipping agent with information on the damage.

For repair or replacement of any Ciprico product damaged in transit call Ciprico for return instructions and authorization.

Unpacking the Board

Before you remove the RF3880 adapter from its box you should observe precautions to prevent damage by static: wear a wrist band and strap while handling the adapter.

Figure 2 Static Precaution

Step 2: Jumpers and Termination

The RF3880 board has four separate jumper blocks, shown in the figure below. You will need to consider each of the jumper blocks to determine whether your application requires changes from the default settings.

Specific locations of the blocks are shown below:

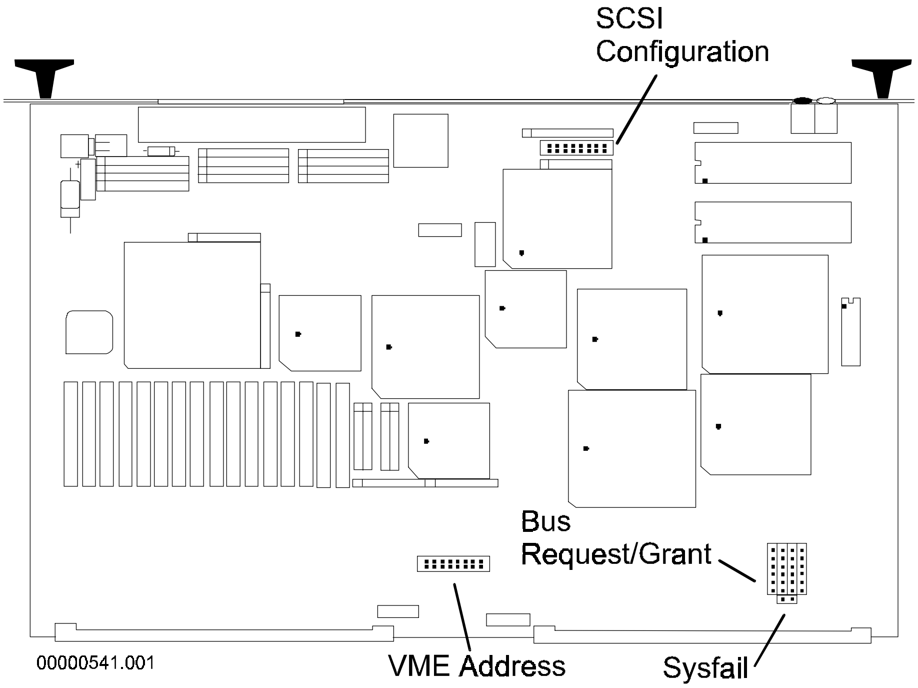


Figure 3 Locations of Rimfire 3880 Jumpers

Each of the jumper blocks shown above will be discussed in the following pages.

SCSI Configuration Jumpers

This block of jumpers is used to select power-up defaults for:

- Adapter SCSI ID
- SCSI Bus reset on power-up and system reset
- SCSI Bus Parity

The SCSI Configuration jumper block is illustrated below:

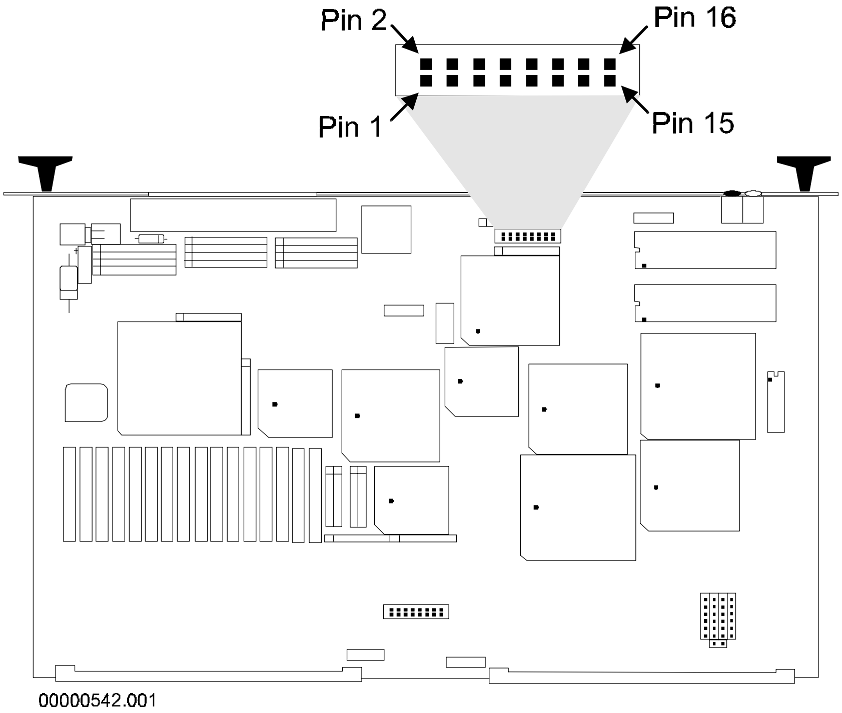


Figure 4 SCSI Configuration Jumper Block

The default settings of the jumper block are shown in the above figure and the table below.

Pins	Description	Default
1-2	Reserved	OUT
3-4		OUT
5-6	Enable SCSI Parity Checking	OUT = Selected
7-8	SCSI bus Reset at Power-up	OUT = Selected
9-10	SCSI ID Bit 3	

Table 13: Default Jumper Settings

Bus Request/ Grant Jumper

This jumper block is used to select the level at which the RF3880 adapter can request and be granted access to the VMEbus by the ARBITER. The four sets of Bus Request and Grant signals are daisy-chained separately on the VMEbus backplane. Only one of the four levels can be used by a single REQUESTER. The Bus Grant signals that are not used are passed through to the next board on the backplane. The Bus Grant signal will be gated on the adapter for whichever level you choose. If the RF3880 adapter has a request for the bus pending, it will take control of the VMEbus. If it has no request pending it will pass the Bus Grant signal to the next board. The Bus Request/Grant jumper block is illustrated below.

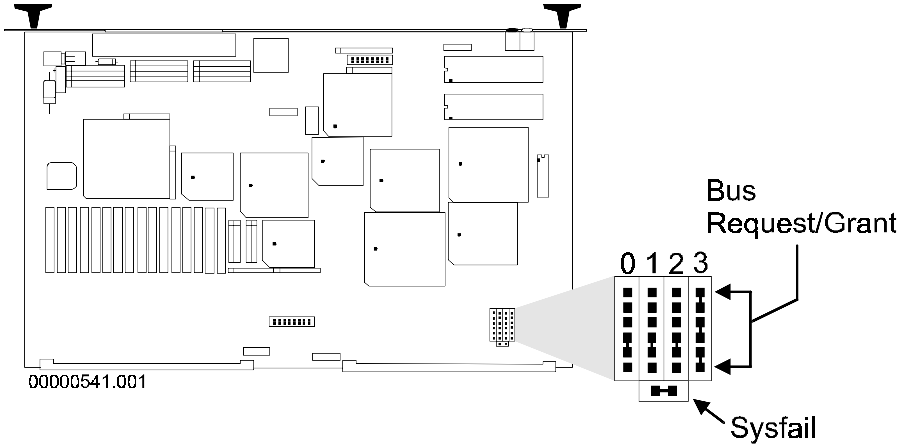


Figure 5 Bus Request/Grant and Sysfail Jumpers

The setting shown in the above figure is the default from the factory. It indicates level 3 for Bus Request/Grant.

To select a Bus Request/Grant level, place three jumpers down the level you wish to use. The unused levels should have jumpers placed from pin 4 to 5.

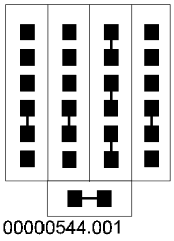


Figure 6 Bus Request and Grant on Level 2

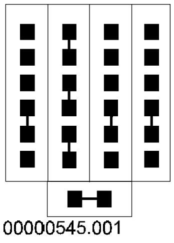


Figure 7 Bus Request and Grant on Level 1

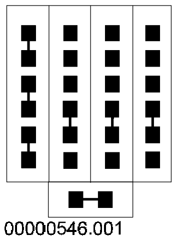


Figure 8 Bus Request and Grant on Level 0

Sysfail Jumper

The VMEbus Sysfail signal is (according to standards) to be asserted during Power-on Self-test to indicate Self-test is occurring. Some systems instead interpret the signal as a failure. For this reason, the RF3880 adapter jumpers the signal on the board.

IN - Sysfail signal asserted on bus during Self-test.

OUT - Sysfail signal not asserted on bus.

Table 5 on page 3 - 7 illustrates the location of the Sysfail Jumper. As shown, the default for this jumper is IN.

VMEbus Address Jumpers

This set of jumpers is used to select the address at which the RF3880 adapter can be addressed by the system. The jumpers set the upper 7 bits of the 16-bit address, thereby reserving 512 bytes of address space.

Also available to be set is bit 2 of the address modifier lines. With this jumper in you can elect to have the RF3880 adapter reserved for short non-privileged access only. If the jumper is out, the board will be in short supervisory access mode.

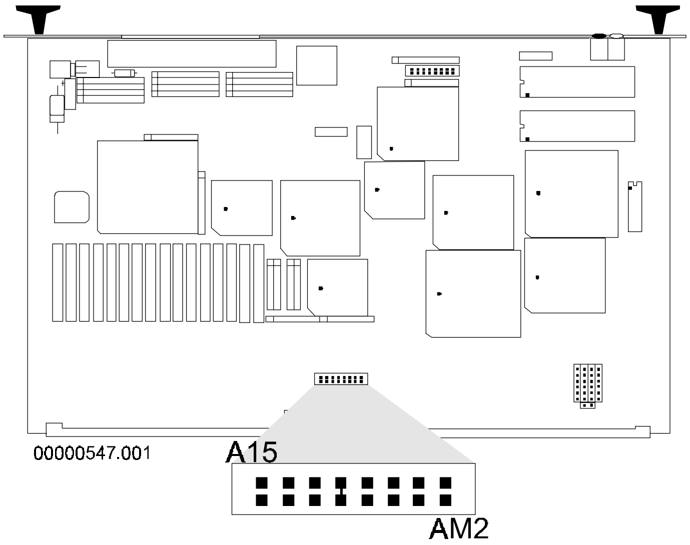


Figure 9 VMEbus Address Jumper

The VMEbus Address jumpers are illustrated in Fig. 3-4. The default address setting of EE00H, with address modifier of 2DH is shown.

For the VMEbus Address Jumpers:

IN - Is equal to a 0 for that address bit.

OUT - Is equal to a 1 for that address bit.

For the Address Modifier Jumper:

IN - Is equal to Address Modifier 29H. (short, non-privileged access)

OUT - Is equal to Address Modifier of 2DH. (short supervisory access)

The VMEbus Address Jumper block pins correspond to the VMEbus address lines as follows:

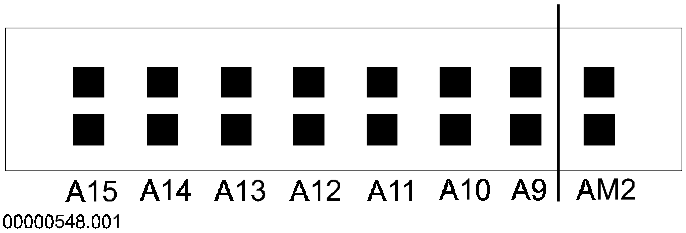


Figure 10 VMEbus Address Jumper Block Pins

Example Addresses

Below is a table of some common RF3880 Base Addresses. The table demonstrates the settings of the VMEbus Address Jumpers and the VMEbus Address Modifier Jumper for each of the addresses.

VMEbus Address Lines								Base Address	Address Modifier
A15	A14	A13	A12	A11	A10	A9	AM2		
Out	Out	Out	In	Out	Out	Out	Out	EE00H	2DH
Out	Out	Out	In	Out	Out	Out	In	EE00H	29H
Out	Out	In	In	Out	In	Out	Out	CA00H	2DH
Out	In	Out	In	In	In	In	In	A000H	29H
In	Out	In	Out	In	In	In	Out	5000H	2DH
In	Out	In	In	In	In	In	In	4000H	29H
In	In	Out	Out	Out	Out	Out	Out	3E00H	2DH
In	In	Out	Out	Out	Out	In	In	3C00H	29H
In	In	Out	In	Out	In	Out	Out	2A00H	2DH
In	In	Out	In	Out	In	In	In	2800H	29H

Table 14: Example Base Address Jumper Settings

SCSI bus Termination

Each end of the SCSI bus cable must be terminated to prevent corruption of the data and control signals. The RF3880 adapter has terminator packs on the board. These are removable in the case that the adapter is not at the end of the SCSI cable.

The adapter provides termination power on the SCSI bus cable. This line is fused on the board. The fuse is replaceable.

The location of the terminators and removable termination power fuse are shown in the following illustrations.

The following figure illustrates the board termination.

Note
Two types of SCSI bus terminators are used on the RF3880 mother board. The first two in each column, closest to the SCSI connector, are 330—ohm SIPs, and the others are 150—ohm SIPs.

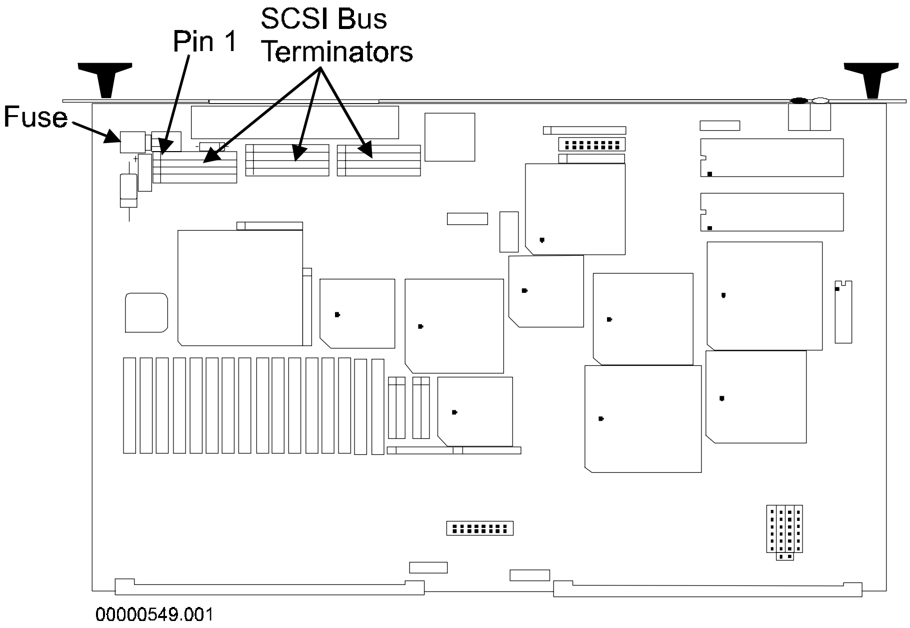


Figure 11 Board Fuse and Terminators

Note
It is very important to replace the termination fuse with an exact equivalent. If you replace the termination fuse with a value lower or higher, you risk continued problems or damage to the RF3880 and attached equipment.

Replacement Fuse

In the event that the termination power fuse is blown, replace it with:

Littlefuse P/N 273-002 or equivalent

Step 3: Board Insertion

Although the VMEbus Specification has very precise mechanical and electrical requirements for the subrack, it still leaves room for variance in the design of the system enclosure. You will need to examine the enclosure that you will be working with to determine how to get access to the subrack.

General Steps

In most cases you will need to follow these steps to insert the board:

1. Remove power from the system.
2. Remove as much of the enclosure shell as is necessary to adequately access the VMEbus subrack.
3. Choose a free double-height slot for the RF3880.
4. Remove bus grant arbitration and interrupt acknowledge jumpers from backplane. You will need to remove the four Bus Grant jumpers, and the Interrupt Acknowledge daisy-chain jumper from the slot chosen for the RF3880. These jumpers must be installed only for empty slots.
5. Align the RF3880 with the board guides on the subrack for the slot you have chosen. Be sure the orientation is correct. (If J1 on the subrack backplane is up, the component side of the board should face right.)
6. Slide the board forward and firmly press into the J1 and J2 connectors on the backplane. You should feel it give slightly as you press inward.
7. If your system subrack accommodates them, fasten the small screws on the RF3880 into the subrack. You will need a small flat bladed screwdriver.

The illustration on the next page demonstrates insertion of the RF3880 into a VMEbus subrack. Do not reinstall the enclosure panels yet. You will need to observe the faceplate LEDs in the next step.

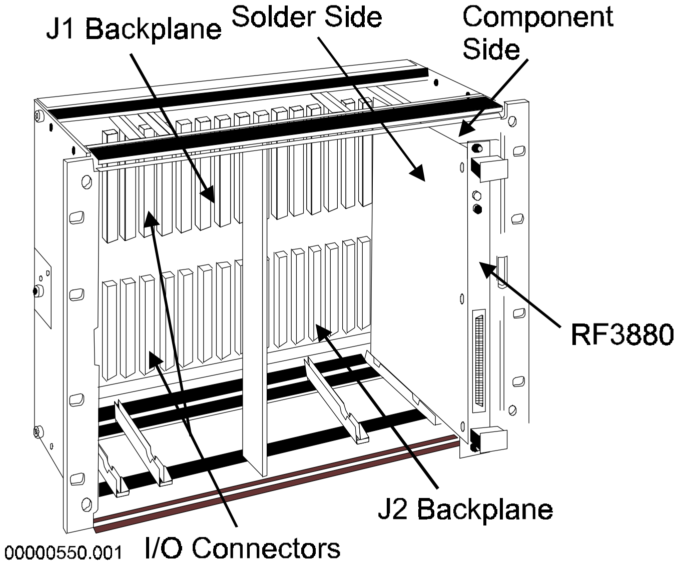


Figure 12 Inserting the RF3880

Step 4: Apply power and observe LEDs

After you have inserted the board into the VMEbus slot, apply power to the system.

There are two LEDs on the faceplate of the RF3880: one is green and the other is amber.

Meaning of LED's

The LED's are used in the following manner:

- The green LED indicates BUSY status when the board is processing commands.
- The amber LED indicates an ERROR status.

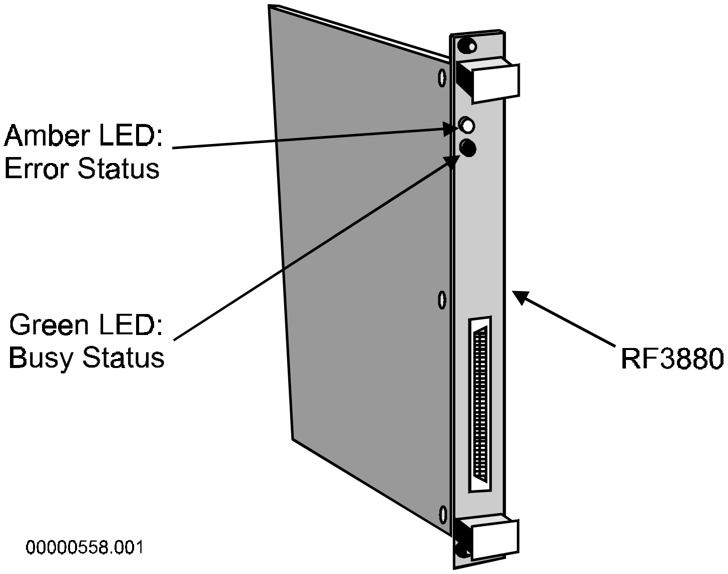


Figure 13 Location of LEDs

Power-up Sequence

At a normal power-up you should observe the following:

1. Both LEDs will briefly light, then the green will go out.
2. The amber LED will stay lit for less than 5 seconds, while self-test is running.
3. The amber LED will go out, indicating self-test has passed.

Error Conditions

The amber LED is used to indicate an error condition. These are the conditions indicated by the amber LED:

BLINKS - This indicates that the adapter has had a self-test failure.

STAYS ON SOLIDLY - This indicates that the board has had a catastrophic error during operation such as bus error or timeout. The Status port will contain an error code that details the type of failure that occurred. These error codes are:

Code	Name	Description
10H	Bad PBIN or SBOUT Value	The Command List PBIN or SBOUT value exceeded the value set for PBNUM or SBNUM.
11H	State wrong for Start/ Stop Command List command	The code is returned under these conditions: **Start Command List** command issued a command list is already active. **Stop Command List** command issued when no command list is presently active. **Command List Channel Attention** issued when no Command List is active.
12H	Target Mode Error	A **Target Mode Selection Area Read** Channel Attention issued when no selection are was written by adapter.
14H	Software VMEbus Timeout	The adapter times each of its VMEbus transfers; it will report this if a VMEbus transfer takes too long to complete. This can occur for transfers of parameters as well as data.
15H	VMEbus Error Occurred	A VMEbus error was detected while the adapter was transferring either parameters or data.
96H	Internal Firmware Error	This error code indicates that a Firmware error was detected during the execution of a command.

Table 15: Status Port Catastrophic Error Codes

Step 5: Configure and Connect SCSI Devices

The RF3880 can communicate with any SCSI device that adheres to the SCSI Common Command Set. Each separate device on the SCSI bus must have a unique SCSI ID and be properly terminated. In addition, it may have a self-test sequence or options and parameters that you should know about.

You will need to examine the documentation that arrives with your selected peripheral for more information. This section outlines what to look for.

Selectable Options

Each peripheral vendor can choose a unique way to assign a SCSI ID and enable/disable options. The two most common methods are: jumpers and switches.

You will need to find the section in your peripheral documentation that explains how to set the options.

SCSI ID

You will need to decide which SCSI ID to assign to your peripheral(s). Bus access priority is given to a device based on the SCSI ID it is assigned. The SCSI ID’s zero to seven have higher priority than eight through 15, and priority descends from seven to zero, and 15 to eight. See the SCSI 3 specification for more information on this. Each device on the SCSI bus (including the RF3880) must have a unique SCSI ID. The factory-set ID for the RF3880 adapter is zero.

Options

Your device may also have a means to select options such as: termination power source, automatic spin-up, or parity checking.

Termination Power: If your hard disk is the last peripheral on the SCSI bus, it will need to have terminators installed. Some hard drives allow you to select whether the terminators will get power locally, on the drive, or use the termination power available on the SCSI cable. Setting up the drive to provide its own terminator power is the most universal solution. The RF3880 provides TERM PWR on the SCSI cable.

Automatic Spin-up: Some disk drives allow you to delay spin-up of the disk media until a SCSI command is issued to spin it up. The advantage to this is that it cuts down the power draw when the system is initially turned on.

Parity Checking: The RF3880 is flexible enough to allow you to use SCSI bus parity checking or disable it. If you wish to use it, parity checking must be enabled on each peripheral and also enabled using the *General Options* command explained in Chapter 7. Note that all devices on the SCSI bus must be setup for parity the same way.

You can select a power-up/reset default for parity checking with pins 5 and 6 of the SCSI Configuration Jumper Block. The value you use in the *General Options* command overrides the jumper setting, so you should be sure that they match. (You would install a jumper over the pins to disable parity checking.) Moreover, you can use the *Board Information Board-control* command to discover the current setting of the parity jumper without removing the board. See Chapter 7 for more information on Board-control commands.

Termination

The devices at each physical end of the SCSI bus cable must have terminating resistors installed. If termination is not correctly applied to the bus, data corruption can occur. If you are using only two devices on the bus (the RF3880 and one peripheral) both must be terminated. If you are using multiple devices on the SCSI bus (several peripherals, in addition to the RF3880 adapter) only the two devices physically at the ends of the cable must be terminated.

Cabling

The SCSI cable connector for the RF3880 extends from the front faceplate. Currently there is one model of the RF3880, and it uses a SCSI “P” type connector to implement the differential interface.

Model	SCSI mode	Max Cable Length	SCSI Connector
RF3886	Differential	25 meters (82 ft.)	68 pin ‘P’ connector

Table 16: SCSI Cable Connector

4

Command Operation

Introduction

The purpose of this chapter is to provide a comprehensive view of the RF3880 firmware operation and how it works with the hardware described in Chapter 2. By explaining the command structures and when to use them, this chapter explains how to use the board and it explains in general terms what a board driver must do.

Summary

This chapter discusses the following:

- Format of the basic Parameter and Status blocks from which command structures are built.
- Types of command structures—Single Command and Command List.
- The process for issuing a single command.
- The process for creating and starting a command list.
- Examples of both types of command structures.
- Implementation of the command structures within a ‘C’ routine.
- Types of commands you can issue—Pass-through and adapter Board-control commands.
- When to use each type of command.

The explanations in this chapter are most helpful to anyone designing a driver for the RF3880 adapter or attempting to understand an existing driver for the RF3500 family of adapters.

What the Adapter must accomplish

There are several basic operational requirements that the RF3880 adapter must fulfill:

The Host must have a method to communicate details of command operation with the adapter — Communication from the host to the RF3880 adapter takes the form of command structures. There are two types of command structures you can use with the RF3880 adapter: Single Command and Command List. The latter is preferable in most situations because it boosts the performance of the adapter. There are instances when you must use the single command method of communication. Both types of structures are described in the following sections.

The adapter must have a means to communicate the outcome of a command process — Reporting the completion status of the command is accomplished via status blocks which are part of the command structure. Just as there are two types of RF3880 command structures for you to use when issuing a command, there are two ways status structures are returned from the adapter. Each uses slightly different protocol to notify the Host of returned status. This is described in more detail in a later section.

The Host must have a way to alter the behavior of the adapter to suit its environment — To meet the versatility required by differing host operating systems and application needs, the RF3880 adapter has a set of board level commands for you to use. These commands allow you to select options relating to the system bus of the host and also tailor operation for the devices attached to the board. They are referred to as Board-control commands.

The Host must have a defined protocol for issuing commands to the adapter (acting as an initiator or a target) — The RF3880 adapter is an interface between the VMEbus and SCSI bus architectures. When the VMEbus Host requests an operation involving the SCSI bus, the RF3880 issues a command to a device on the bus, using SCSI protocol, or acts as the intermediary for an initiator on the bus. The RF3880 uses the defined SCSI command structure within its own parameter-block command structure. This allows the RF3880 to pass commands to the SCSI device without modification (in Initiator Mode) or receive a SCSI command and pass it to the Host for interpretation (in Target Mode). These are called Pass-through commands.

The adapter must provide status from a device regarding the command just completed by the device — Just as the defined SCSI command structure is part of the RF3880 command structure and is passed through to the device, the Sense Bytes from the SCSI device are reported to the host within the Status Block returned by the adapter.

Each of these operational requirements is discussed in more detail on the following pages.

The Base Parameter Block

The Parameter Block is one of the basic building blocks of the RF3880 command structures. It is designed to hold all facts relevant to a single command.

The same Parameter Block format is used whether the command is a Pass-through command directed to a SCSI device, or a Board-control command to be interpreted and acted on by the adapter. (The various flags and byte values may have different meanings in each case though.)

All Board-control commands are described in detail in Chapter 7. More information about how the RF3880 implements SCSI commands can be found in Chapter 5. See a SCSI specification and vendor-supplied peripheral documentation for details about particular SCSI commands.

Parameter Block Format

The Parameter Block format for all Pass-through and also adapter Board-control commands is illustrated below:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Command Options, Addresses, and Transfer Count			Target/Adapter ID
08H				
0CH				
10H	Command Code	Command Descriptor Block		
14H				
18H				

Table 17: Base Parameter Block

A brief description of each of the fields follows:

Command Identifier

When setting up a command structure, you use this field to distinguish a command in order to associate it with a returned Status Block. The value is a double-word and must be unique; the Command Identifier for each currently executing command should be different. Any unique four-byte value will work. You could, for example, use the command address or a Parameter Block Index pointer as the Command Identifier. Both Pass-through commands and adapter Board-control commands use this field for proper identification of status to be returned whether the command is issued in a Single Command structure or a Command List.

Command Options, Addresses, and Transfer Count

You will use these fields of the Parameter Block to supply an address and transfer count when a data transfer is required by the command you wish to issue. There are also some command-modifying options available to you depending on the command you are issuing.

Target/Adapter ID - The Target/Adapter ID byte is very important to the command issue process. The value you supply here determines whether the command described in the Command Descriptor Block is a Pass-through command or an adapter Board-control command:

- If the ID = FFH, the command is a Board-control command.
- If the ID = 0-FH (0-15), the command is destined for the SCSI target or initiator with that ID.

Command Descriptor Block

In this part of the Parameter Block you must supply details about the command to be executed. The Command Descriptor Block will contain a SCSI command, a Target Mode response, or an adapter Board- control command. The RF3880 Board-control commands are explained in detail in Chapter 7. You can find more information about issuing Pass-through commands later in this chapter and in Chapters 5 and 6. For detail about actual SCSI commands, see a SCSI specification or your vendor-supplied peripheral documentation.

Command Code - This field will always contain a command code for either a Pass-through or an adapter Board-control command.

The Base Status Block

The Status Block is another of the building blocks of the command structures you will be using when issuing commands. Used in conjunction with the Parameter Block, its purpose is to report the outcome of a command you issued.

The Status Block format is the same whether you have issued a Pass-through command or a Board-control command. It appears the same whether the command structure you used is a Single Command or a Command List, however there is a difference in the number of Status Blocks you may receive for each type of command structure.

Single Command structures return only one Status Block. The Single Command structure will contain only the last Status Block—the Status Block with the CC (Command Complete) bit of the Flags byte set.

When you are using a Command List structure, multiple Status Blocks for any one command are possible depending on the retry and Sense Byte selections you have made (with the Unit Options or Extended Unit Options Board-control commands).

Status Block Format

The Status Block Format you will see used for all returned status information is illustrated in the following figure:

Address Offset	Byte Memory Offset			
	Offset + 0	Offset + 1	Offset + 2	Offset +3
00H	Command Identifier			
04H			Error	Flags
08H	Status Information			
0CH				

Table 18: Base Status Block

Following is a brief description of each of the fields:

Command Identifier - This value will identify which Parameter Block has completed.

Flags

This byte can be polled to determine whether, and in what condition, a command completed. It has this format:

BITS							
7	6	5	4	3	2	1	0
CC	ERR	RTY	DTT	DTG	CSB	0	TMS

Table 19:

TMS Target Mode Status - This bit indicates that Status is being returned from the Target Mode Response command.

0 ⇒ The status was not a result of a Target Mode Response command.
1 ⇒ The status is the result of a Target Mode Response command.

CSB Continued Status Block - This bit indicates that the current Status Block is one of a series of Status Blocks generated because extended Sense Bytes are enabled. Only valid for Pass-through commands issued through Command List.

0 ⇒ Status Block is first (or only) Status Block
1 ⇒ Status Block is one of a series of Status Blocks.

DTG Data Transfer Greater - Indicates when a Initiator Pass-through command completes with more bytes requested by the SCSI device than was specified in the transfer count field.

0 ⇒ Data transfer was not greater.
1 ⇒ Data transfer was greater.

DTT Data Transfer Truncated - Indicates when a Initiator Pass-through command completes with fewer bytes transferred than requested.

0 ⇒ Data transfer was not truncated.
1 ⇒ Data transfer was truncated.

RTY Retry - Indicates whether a Pass-through command required retries to complete.

0 ⇒ No retries were required.
1 ⇒ Retries were required to complete.

ERR Error Status - Identifies when a Board-control or Pass-through command completes with an error.

0 ⇒ No error occurred.
1 ⇒ An error occurred.

CC Command Complete - Indicates whether a Board-control or Pass-through command has finished.

0 ⇒ Command not complete.
1 ⇒ Command complete.

Error - When an adapter-detected error occurs, this byte contains an error code to describe the condition. Error Codes are described in Appendix A.

Status Information - Depending on the type of command that was issued, the RF3880 will return different types of Status Information in these fields. For Initiator Pass through commands, SCSI Status and Sense Bytes will be returned. For details see Chapter 5. Target Mode Response command Status Blocks are described in Chapter 6. See Chapter 7 for details about any special Status Blocks returned from Board-control commands.

Multiple Status Blocks

It is possible to receive more than one Status Block to indicate the outcome of a command. To receive multiple Status Blocks, when operating in Initiator Mode, all the following must be true:

- You have used the Unit Options or Extended Unit Options Board-control commands to set up retries and / or extended Sense data for Pass-through commands.
- The command is a Pass-through command. (Board-control commands are not subject to retries and do not return Sense Data.)
- The Pass-through command is issued through use of the Command List structure. (Single command structures can only contain one Status Block.)

It is also possible to receive multiple Status Blocks when operating in Target Mode. You will receive two Status Blocks in Target Mode if the SCSI Command Descriptor Block is more than six bytes. See Chapter 6 for more information.

Setting up Unit Options

These are the options selected through the Unit Options and Extended Unit Options Board-control commands that cause the Rf3880 to return multiple Status Blocks in response to an error condition:

Retries - If you 1.) supply a Retry Limit value, 2.) select a type of retry in the Retry Control field, and also 3.) set the ISB bit of the same field, each command retry performed will generate at least one Status Block.

Sense Data - If you select a Sense Count greater than eight, or specify more than eight Selected Sense Bytes, you will receive multiple Status Blocks for each Request Sense command sent in response to Check Condition.

Note
If you enabled both Retries and nine or more byte of Sense Data, you will receive multiple Status Blocks for every retry performed until either the command completes successfully or the Retry Limit is reached. This can be calculated as follows:

For a command that exhausts Retry Limits:

> # of Status Blocks = (# of retries + 1)x(# of Status Blocks needed for Sense Data)

For a command that recovers before Retry Limit is reached:

> # of Status Blocks = ((# of attempts + 1)x # of Status Blocks needed for Sense Data) + 1

Single Command Structure

There are two ways to issue a command to the RF3880 adapter. The simplest method involves the use of the Single Command structure. This 52-byte structure combines a Parameter Block and a Status Block into a single structure along with a field that specifies an interrupt level and ID. This Single Command structure is shown below. The important things to remember about the Single Command structure are:

- It is designed to be used for initial set-up.
- The first command you issue to the board after power-up or reset must use this structure.

Format of Single Command Structure

The Single Command structure is illustrated for you below:

Address Offset	Byte Memory Address			
	Offset +0	Offset +1	Offset +2	Offset +3
00H	Parameter Block (detail for Initiator Pass-through commands on page 5 - 3, see Chapter 7 for details on individual Board-control commands.)			
04H				
08H				
0CH				
10H				
14H				
18H				
1CH	Reserved		Reserved	
20H	Reserved			
24H	Status Block (detail for Initiator Pass-through status on page 5 - 12, see Chapter 7 for detail on individual Board-control status.)			
28H				
2CH				
30H				

Table 20: Single Command Structure

Parameter Block

This section of the Single Command structure is where you build a Parameter Block containing information related to the command you wish to execute: Initiator Pass-through or Board-control.

Most of the fields are command-specific; if there is data to transfer, for example, an address, address modifier and transfer count must be provided. For details about the parameter block fields necessary for a Initiator Pass-through command refer to Chapter 5, Initiator Pass-through Commands and a SCSI specification. For details about a Board-control command see Chapter 7, Board-control Commands.

Reserved Fields

Reserved fields must be zeroed. If a non-zero value is permitted in a Reserved block, you may get an error.

Interrupt

This field selects the Interrupt level which the RF3880 adapter will use when the command issued through the Single Command structure completes. You can also denote the resultant interrupt acknowledge ID that the adapter will place on the data bus in response to the Host. The Interrupt field uses these bit-meanings:

BITS															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	INT			ID							

Table 21: Bit-meanings of Interrupt Field

ID Status/ID - This byte denotes the Status/ID value that the adapter will provide to identify the interrupt being acknowledged by the Host.

INT Interrupt Level - You may elect to use any one of the following levels for adapter interrupt:

000 — Interrupt disabled
001 — Interrupt level 1
010 — Interrupt level 2
011 — Interrupt level 3
100 — Interrupt level 4
101 — Interrupt level 5
110 — Interrupt level 6
111 — Interrupt level 7

Status Block - The Status Block is the means by which the RF3880 adapter communicates successful command completion, error conditions, retry information or SCSI Sense Bytes. Status Block information specific to the type of command you issued can be found in Chapters 5, 6 and 7.

Note
If you have set the ISB (Issue Status Block) bit in the Retry Control field of the Unit Options or Extended Unit Options command, it will be ignored for commands issued through Single Command structure.

If the status from a command issued within a Single Command structure requires multiple Status Blocks to report the status, only the last block is saved in the structure.

Using a Single Command Structure

The Single Command structure is the fundamental means by which you can issue a command to the RF3880 adapter board. To issue a command you must perform each of these steps in the order described.

Set-up the Structure in memory

1. Clear 52 bytes of system memory at the location you wish to create the Single Command structure.

2. Build a Parameter Block in bytes 0 through 28. Refer to Chapter 5 for more information about Initiator Pass-through commands. Refer to Chapter 7 if you wish to issue a Board-control command like Start Command List.

3. Assign a value to the Interrupt field of the Single Command structure, reflecting the interrupt the adapter should use and the ID by which it will identify the interrupt being acknowledged by the Host. The Reserved fields must be zero.

Prepare the Hardware Ports

4. If you are going to use this command to set bus width and byte/word swapping, determine the appropriate value for the Control Field of the Address Buffer Port.

5. Write the Control Field, Address Modifier and the two address words of the Single Command structure to the Address Buffer Port on the adapter.

6. Read the Status Port. Test the RDY bit (bit 1) to confirm that the board is running. Next, verify that bit 0 (ENT) indicates the board is ready for a Single Command. (On reset or power-up, this bit is set to zero. Thereafter it toggles from its previous value to indicate readiness.)

Execute the Command

7. Write a 0 value to the Channel Attention Port.

Determine Command Completion

8. When the command is complete, the adapter will write a Status Block in bytes 36 through 52 of the Single Command structure and generate any interrupt you requested. If you did not specify an interrupt level in the structure, you must poll the CC bit of the Flags byte of the Status Block to determine completion. Poll the ENT bit of the Status Port to determine when you can issue another Single Command. Command pre-fetch allows you to send another command to the adapter before the previous is completed. Therefore, the CC and ENT bits are not equivalent.

Some Example Single Commands

The following examples will help you understand how to use the Single Command structure. Two examples are given here. The first is an example of a Board-control command. The second shows you how to issue a Initiator Pass-through command to the RF3880 adapter.

Example 1: Start Command List Board-control command

The Start Command List Board-control command sets up a Command List structure for you to use for subsequent commands. (Basically, the Command List uses a location in memory as a depot for multiple commands and their Status Blocks. This feature is discussed in detail later in this chapter.)

This example shows you how to use the Single Command Structure to initiate the Command List.

1. Select a portion of system memory for the Single Command structure. You will need 52-bytes of space to hold the Parameter Block, Interrupt and Reserved fields, and the Status Block that make up a Single Command structure.

   Example Memory Address = 822F4H

2. Fill in the Parameter Block information. (Detailed Parameter Block information for each Board-control command is given in Chapter 7.)

   These are the Parameter Block values for this example:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier = 000822F4H			
04H	Reserved		Addr Mod = 3DH	ID = FFH
08H	Command List Memory Address = 00099450H			
0CH	Reserved		Interrupt = 0300H	
10H	Command = 10H	Reserved		
14H	Reserved			
18H	Reserved			

Table 22: Example Start Command List PB

Command Identifier = 822F4H

Even though the Status Block is part of the Single Command structure and easily identified it is good to provide a value here in order to be consistent with Command List structures. The value chosen is the address of the Single Command structure.

Address Modifier = 3DH

This value indicates that Standard Supervisory Data Access will be used for Command List data transfers.

ID = FFH

Indicates to the RF3880 adapter that this is a Board- control command.

Command List Memory Address = 99450H

System Memory address of the Command List structure.

Interrupt = 0300H

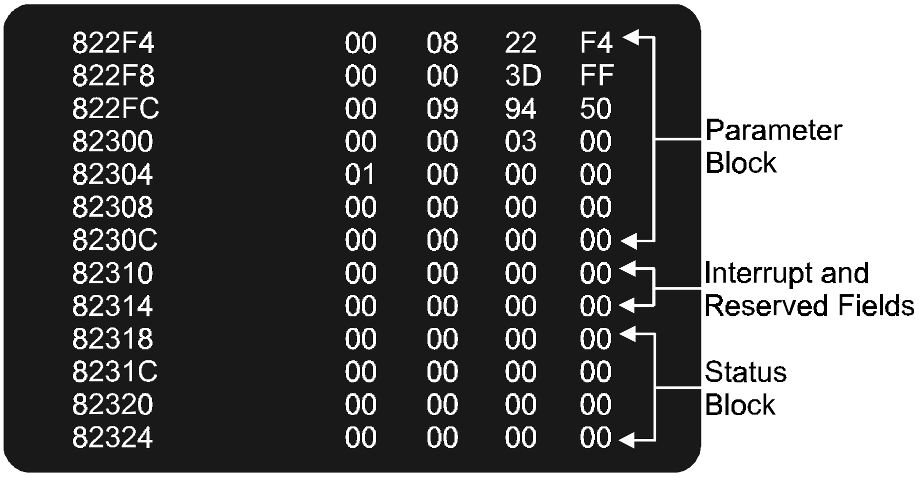
Indicates that an Interrupt Level of 3 be used for Command List status.

Command = 01H

This value is the Start Command List command code.

3. Fill in the Interrupt Word of the Single Command structure. This is used to indicate the Interrupt Level to be used for the Single Command. In this example we will leave it zero along with the Interrupt Acknowledge ID. This means the Flags byte of the Status Block will need to be cleared before issuing the command, and then the Command Complete (CC) bit polled to determine when this command completes.

   In memory, the structure will look like this:



```
822F4    00   08   22   F4
822F8    00   00   3D   FF
822FC    00   09   94   50
82300    00   00   03   00
82304    01   00   00   00
82308    00   00   00   00
8230C    00   00   00   00
82310    00   00   00   00
82314    00   00   00   00
82318    00   00   00   00
8231C    00   00   00   00
82320    00   00   00   00
82324    00   00   00   00
```

00000552.001

Figure 14 Single Command Structure in memory

4. The next step is to load the Address Buffer Port. This will require three writes to the port address. Each of the three writes is described next:

Control Field = 84H - For the first write, the Control Field, we’ll select no byte or word swapping of control structures, but choose to use a 32-bit system bus.

Address Modifier = 3DH - This is the Address Modifier that the adapter is to use when reading the Single Command Structure.

Single Command Structure Address = 822F4H - The other two writes to the port contain the address of the Single Command Structure.

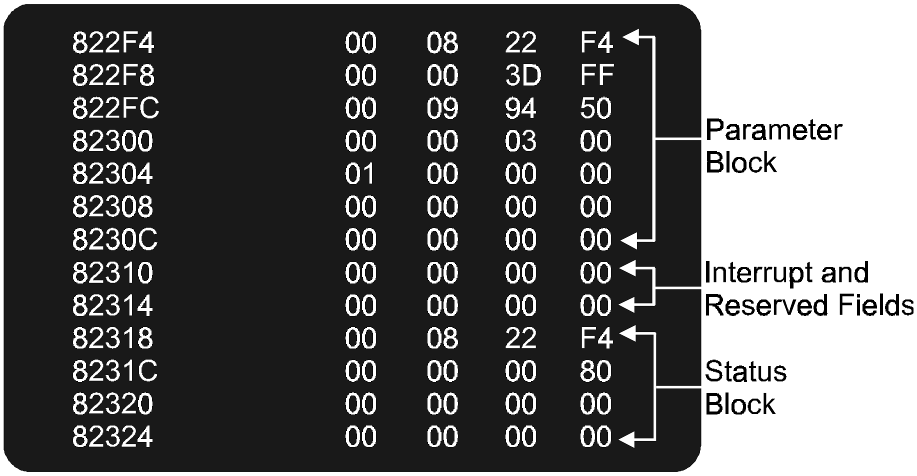
5. The example board is jumpered at address EE00H. The three writes to the Address Buffer Port at address EE00H will be as follows:

   1.) 843DH
   2.) 0008H
   3.) 22F4H

6. Read the Hardware Status Port at address EE10H. Since the ENT bit of the Status Port toggles between 0 and 1 with each Single Command Channel Attention issue, the value returned could be 0202H or 0203H depending on the previous value. We’ll assume this is the first command after a reset; the initial value of the ENT bit is 0. The adapter is ready if the Status Port reads 0202H.

7. To execute the command, write a 0 to the Channel Attention Port.

8. Poll for command completion by reading the Flags Byte of the Status Block and comparing it to zero. When the value is non-zero, the command is complete. The value of the Flags byte in the Status Block will indicate if the command completed with an error. In this example, the command was successful; a value of 80H is returned as shown below. A returned value of C0H would indicate an error had occurred.



```
822F4     00   08   22   F4
822F8     00   00   3D   FF
822FC     00   09   94   50
82300     00   00   03   00
82304     01   00   00   00
82308     00   00   00   00
8230C     00   00   00   00
82310     00   00   00   00
82314     00   00   00   00
82318     00   08   22   F4
8231C     00   00   00   80
82320     00   00   00   00
82324     00   00   00   00
```

00000553.001

Figure 15 Single Command Structure in memory

Example 2:Inquiry Pass-through Command

This example describes the issue of a Initiator Pass-through command. The SCSI Inquiry command is fully described in the SCSI specification.

This example shows you how to use the Single Command structure to issue an Initiator Pass-through command.

1. The set-up of a Pass-through command is the same as for a Board-control command. You must select a portion of system memory for the Single Command structure. You will need 52-bytes of space to hold the Parameter Block, Interrupt and Reserved fields, and the Status Block that make up a Single Command structure. For this example we'll use the same memory address.

   Example Memory Address = 822F4H

2. Fill in the Parameter Block information. (Detailed Parameter Block information for Initiator Pass-through commands is given in Chapter 5. For information about the SCSI Inquiry command, you must reference a SCSI specification or drive manual.)
   These are the Parameter Block values for this example:

Address Offset	Byte Memory Address				
	Offset + 0	Offset + 1	Offset + 2	Offset + 3	
00H	Command Indentifier = 000822F4H				
04H	Flags-2 = 0	Flags-1 = 0	Addr Mod = 3DH	ID = 01H	
08H	VME Memory Address = 00099450H				
0CH	Transfer Count = 24H				
10H	Command = 12H	LUN = 0	Reserved = 0	Reserved = 0	Inquiry 6-Byte Command Descriptor Block
14H	Allocation = 24H	Flag/Link = 0	0	0	
18H	0	0	0	0	

Table 23: Example Pass-through command PB

Command Identifier = 822F4H - Even though the Status Block is part of the Single Command structure and easily identified it is good to provide a value here in order to be consistent with Command List structures. The value chosen is the address of the Single Command structure.

ID = 01H - Indicates to the RF3880 adapter that this Pass-through command should be directed to device ID 1.

Address Modifier = 3DH - This value indicates Standard Supervisory Data Access mode will be used to execute the Inquiry command.

Flags-1 = 0 - This byte is used to control aspects of command operation. None of the features are enabled for this command.

Flags-2 = 0 - This byte offers further command control options. None of the additional features are enabled for this example.

VME Memory Address = 99450H - System Memory address to which the Inquiry data will be returned.

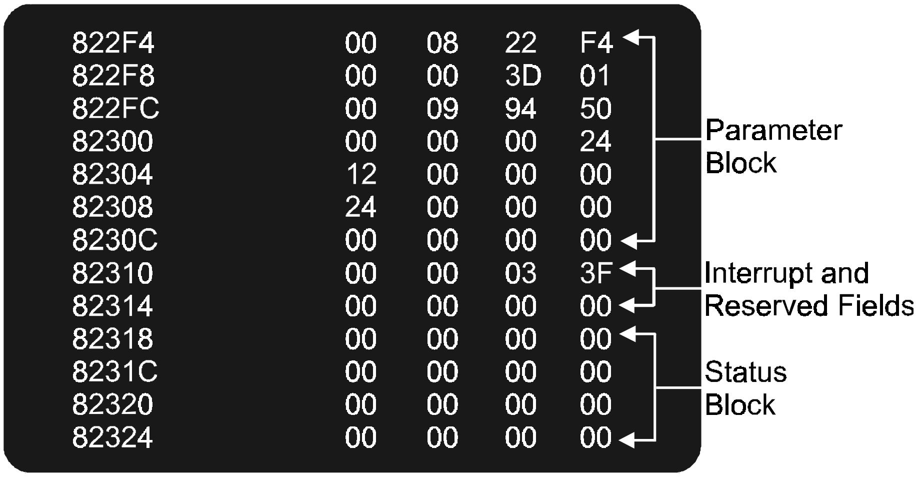
Transfer Count = 24H - This value is the number of bytes of data that will be transferred for this command. The Inquiry data returned by this device is contained in 36 (24H) bytes.

SCSI 6 byte Command Descriptor Block - These six bytes of command information are explained in the SCSI specification.

3. Fill in the Interrupt Word of the Single command Structure. The Interrupt Word is used for the Single Command Interrupt Level.

   In this example we'll use an interrupt level of 3, and designate a Status/ ID of 3FH. This means that the adapter will interrupt the Host when the command has completed by driving IRQ3. When requested for its Status/ID during the VME interrupt cycle, the adapter will respond with the value 3FH on lines D0-D7.

   In memory, the structure will look like this:



```
822F4    00  08  22  F4
822F8    00  00  3D  01
822FC    00  09  94  50
82300    00  00  00  24
82304    12  00  00  00
82308    24  00  00  00
8230C    00  00  00  00
82310    00  00  03  3F
82314    00  00  00  00
82318    00  00  00  00
8231C    00  00  00  00
82320    00  00  00  00
82324    00  00  00  00
```

00000559.001

Figure 16 Single Command Structure in memory

4. The next step is to load the Address Buffer Port. This will require three writes to the port address:

   Control Field = 84H - For the first write, the Control Field, we'll select no byte or word swapping of control structures, but choose to use a 32-bit system bus.

   Address Modifier = 3DH - This is the Address Modifier that the adapter is to use when reading the Single Command Structure.

   Single Command Structure Address = 822F4H - The other two writes to the port contain the address of the Single Command Structure.

5. The example board is jumpered at address EE00H. The three writes to the Address Buffer Port at address EE00H will be as follows:

   1.) 843DH

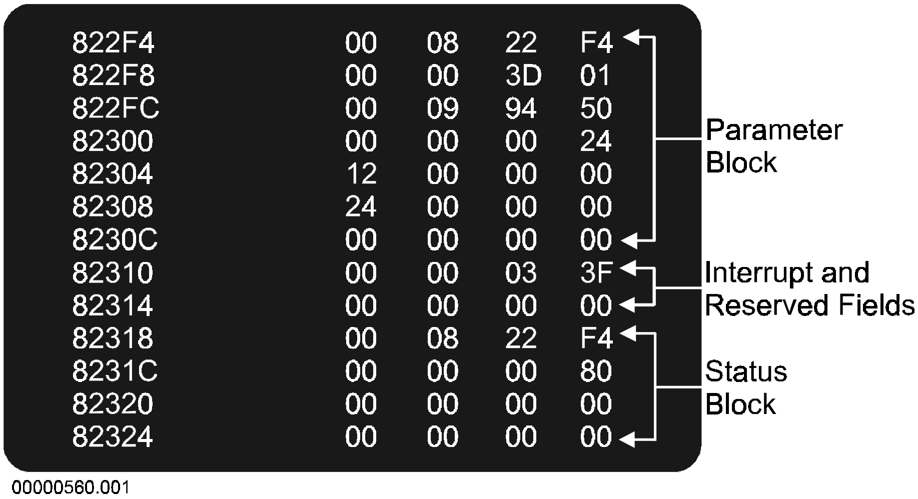
   2.) 0008H

   3.) 22F4H

6. Read the Hardware Status Port at address EE10H. Since the ENT bit of the Status Port toggles between 0 and 1 with each Single Command Channel Attention issue, the value returned could be 0202H or 0203H depending on the previous value. We'll assume this is the first command after a reset; the initial value of the ENT bit is 0. The adapter is ready if the Status Port reads 0202H.

7. To execute the command, write a 0 to the Channel Attention Port.

8. When the command has completed, the Status Block will be written to memory and an interrupt generated by the RF3880 adapter. Read the Flags byte of the Status Block to determine if there are errors, or retries associated with the command.



```
822F4     00  08  22  F4
822F8     00  00  3D  01
822FC     00  09  94  50
82300     00  00  00  24
82304     12  00  00  00
82308     24  00  00  00
8230C     00  00  00  00
82310     00  00  03  3F
82314     00  00  00  00
82318     00  08  22  F4
8231C     00  00  00  80
82320     00  00  00  00
82324     00  00  00  00
```

Figure 17 Single Command Structure in memory

Command List Structure

The previous pages have discussed the use of a Single Command Structure to issue a command to the RF3880 adapter. This section deals with the other method of operation: the Command List Structure. With this structure you are able to queue commands for execution; use of the Command List Structure enhances performance by decreasing command overhead and allowing both system and adapter to perform command I/O at the same time.

Format of Command List Structure

The address of the Command List Structure should be on a double word boundary. The Command List Structure is illustrated for you below:

Address Offset	Byte memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Parameter Block IN Index			
04H	Parameter Block OUT Index			
08H	Status Block IN Index			
0CH	Status Block OUT Index			
10H	Number of Parameter Blocks (n) (max = 2340)			
14H	Number of Status Blocks (y) (max = 4096)			
18H	Reserved			
1CH	Reserved			
20H	Parameter Block #1			
3CH	Parameter Block #2			
.	Parameter Blocks (up to #2340)			
.	Status Block #1			
.	Status Block #2			
.	Status Blocks (up to #4096)			

Table 24: Command List Structure

Description of the Fields

The following section describes the information you will need to properly complete a Command List Structure.

Parameter Block IN Index

This field contains an index into the Parameter Block array. It points to the next open block. When you add a Parameter Block to the Command List Structure, you must increment this number. This value is never changed from the adapter side of operations.

Parameter Block OUT Index

This field also contains an index into the Parameter Block array, but it points to the next Parameter Block in the structure to be read-in by the adapter for execution. This value will be incremented by the adapter after it reads a new Parameter Block; the value should never be changed from the Host side.

Status Block IN Index

The Status Block IN field is incremented each time a Status Block is placed into the Status Block array by the adapter. This value should never be changed from the Host side.

Status Block OUT Index

The Status Block OUT Index advances when the Host reads a Status Block from the array to indicate that a Status Block has been read. This value is never changed by the adapter.

Number of Parameter Blocks

This value indicates the maximum number of Parameter Blocks that you can have active in the list. You should pick a number that is adequate for the number of commands that you will issue. Since the Status Block array follows the Parameter Block array, you cannot increase this number once execution begins. The maximum space available to use for Parameter Blocks is 65536 bytes. Since each Parameter Block is 28 bytes long, the maximum number of Parameter Blocks you can have is 2340.

0 and 1 are not valid values to use in this field.

Number of Status Blocks

This value indicates the number of Status Blocks available to be written by the adapter in the Command List. Maximum space available for Status Blocks is 65536 bytes. Since each Status Block is 16 bytes long, the maximum number of Status Blocks you can have is 4096.

0 and 1 are not valid values to use in this field.

Note
Multiple Status Blocks generated from retries or extended Sense data must be factored into the value you select for this field. Retries and additional Sense data options are enabled through use of the Unit Options or Extended Unit Options commands.

Parameter Block Space

You may have up to 2340 Parameter Blocks in the Command List Structure simultaneously. The Parameter Block area is treated as an independent circular list. First you place a Parameter Block into this area, then the adapter reads it to execute.

Status Block Space

You may have up to 4096 Status Blocks in this area simultaneously. The Status Block area is treated as a circular list separate from the Parameter Block area. When the adapter places a Status Block into this area, it is available to be read in from the Host side.

It is possible to generate multiple Command List Status Blocks from one Pass-through command if retries are enabled and more than eight bytes of Sense data are selected. Retries and Sense data are enabled with the Unit Options or Extended Unit Options commands.

Creating the Command List Structure

To create a Command List Structure you should perform the following steps in the order indicated:

1. Select and clear a portion of memory that will contain the Command List. The Command List should start on a double word boundary. Enough space should be allocated to accommodate the Index information plus the Parameter and Status Blocks. Use the following information to determine the amount of space to allocate. (The value n is the number of Parameter Blocks and the value y is the number of Status Blocks to be held in each respective array.):

   Index information = 32 bytes
   Parameter Block array = n x 28 bytes
   Status Block array = y x 16 bytes

Zero the indexes and set-up the Number of Parameter Blocks and Number of Status Block fields.

2. Issue a Start Command List command to the adapter within a Single Command Structure. An example of how to do this can be found on page 4 - 12.

3. Create Parameter Blocks in the reserved space, update the Parameter Block IN index. Write a 1 to the Channel Attention Port to begin execution.

4. Retrieve Status Blocks from the Status Block reserved area. Update the Status Block OUT index.

Using a Command List Structure

The Command List Structure is a fast method of sending commands to the RF3880 adapter. This section explains usage of the indexes. It also details command issuing procedures.

Index Guidelines

These guidelines explain the use of the Command List Indexes:

- The respective IN and OUT indexes for the Parameter and Status Blocks control each respective circular list. The number placed in each index points to an element in an array: a 0 is the first element in an array; 1 is the second element in an array; etc.
- The IN indexes indicate the next free block of the array to be written from the Host side (Parameter Block) or from the adapter side (Status Block).
- The OUT indexes point to the next block of an array to be read in to the Host side (Status Block) or to the adapter side (Parameter Block).
- The IN and OUT Indexes are manipulated with standard circular list algorithms; If IN is equal to OUT, the list is empty; if IN is equal to OUT minus one (modulo list size), the list is full.
- The Host side is responsible for updating the Parameter Block IN Index and the Status Block OUT Index.
- The RF3880 adapter updates the Parameter Block OUT Index and the Status Block IN Index. These indexes must not be changed from the Host side.
- The Parameter Block OUT Index is not necessarily changed at the time a command is accepted or for each command read onto the board. To reduce system bus activity the adapter performs this action only periodically to keep the list from filling up. Therefore, change in the OUT Index cannot be used to signal command acceptance.

Command Guidelines

Use the following guidelines when issuing commands via Command List:

- Several commands can be loaded into the Parameter Block array at once and issued with a single write of 1 to the Channel Attention Port.
- Similarly, the adapter can return several Status Blocks at once; there may not be an interrupt received for each Status Block. However, you may also receive several interrupts in a row.
- The issue of commands and receipt of status occurs asynchronously; there are no timing restrictions on either.
- The number of possible Parameter and Status Blocks is indicated by the two counter fields: Number of Parameter Blocks; and Number of Status Blocks.

Adding a Parameter Block

To add an entry to the Parameter Block array, follow these steps:

1. Compare the Parameter Block IN Index to the OUT Index to be sure the array is not full.
2. Calculate the offset for the Parameter Block indicated by the Parameter Block IN Index and build a Parameter Block containing the command you wish to issue.
3. Increment the Parameter Block IN Index.
4. Write a 1 to the Channel Attention Port.

Removing a Status Block

To remove an entry from the Status Block array, follow these instructions:

1. Compare the Status Block IN Index with the OUT Index to find out if the list is empty.
2. Read in the Block indicated by the Status Block OUT Index.
3. Increment the Status Block OUT Index.

Example ‘C’ routines for Command List handling begin on the next page.

Example ‘C’ Routines for Command List

The following ‘C’ listing is included here as an example of how you may set-up the Parameter Block, Status Block and Command List structures and manipulate the Command List Indexes.

```
/* Global Defines: */
#define EMPTY   0

#define NPB     100  /* NPB - Num of Parameter Blocks in Command List. */
#define NSB     200  /* NSB - Num of Status Blocks in Command List. */

#define BYTE    unsigned char      /* an 8 bit value */
#define WORD    unsigned int       /* a 16 bit value */
#define DWORD unsigned long        /* a 32 bit value */

/* parameter_block - a structure used to pass a command to the RF3880 VME to
* SCSI adapter. The fields are:

*/

typedef struct
    {

            DWORD    identifier;   /* identifier - command identifier. */

            BYTE     flags-2;      /* command flags. */

            BYTE     flags-1;      /* command flags. */

            BYTE     modifier;     /* modifier - address modifier. */

            BYTE     target;       /* target - target ID or adpater ID. */

/* possible values for flags-1 */

#defineCF_VALID                    0x80  /* dat and dir bits valid. */
#defineCF_FREE                     0x40  /* reserved bit, not used. */
#defineCF_INHIBIT_CMD_COMPLETE0x20  /* command list, synch control. */
#defineCF_FREE1                    0x10  /* reserved bit, not used.*/
#defineCF_IRS                      0x08  /* inhibit request-sense. */
#defineCF_DATA                     0x04  /* data for this opration. */
#defineCF_DIR                      0x02  /* direction; 1 = (writes). */

/* possible values for flags-2 */

#defineCF_JUST_SEND_MSG            0x08  /* message-only operation. */
#defineCF_INHIBIT_DISCONNECT       0x04  /* inhibit disconnect. */
#defineCF_TAG_MESSAGE_TYPE         0x02  /* tag message type. */

       union
       {
          WORD       w[2];
          DWORD      d;

       } sdp;                        /* sdp - source/dest. memory address */

       union
       {
          WORD       w[2];
          DWORD      d;
       } bytes;                      /* bytes - data transfer count */
```

```
        BYTE   cdb[12];                  /* cdb - Command Descriptor Block/
} parameter_block;

/* status_block - a structure used to hold the status information that is returned from
* the adapter.
*/
typedef struct
        {
            DWORD     sb_ident;          /* sb_ident - status block identifier. */

            BYTE      sb_res;            /* sb_res - reserved field. */

            BYTE      sb_status;        /* sb_status - command SCSI status type. */

            BYTE      sb_error;          /* sb_error - error code number. */

            BYTE      sb_flag;           /* sb_flag - command completion flag. */

/* possible values for sb_flag */
#define     ST_CDONE              0x80    /*command complete. */
#define     ST_ERROR              0x40    /*command error. */
#define     ST_RETRY              0x20    /*retry required. */
#define     ST_TRUNCATED          0x10    /* data transfer truncated. */
#define     ST_CONTINUED          0x04    /* block continued from previous. */

            BYTE    sb_info[8]        /* sb_info[8] SCSI request sense info. */

        } status_block;

/* clist - a Command List structure
*/
typedef struct
        {
            DWORD     pbin;              /* parameter block IN pointer. */

            DWORD     pbout;             /* parameter block OUT pointer. */

            DWORD     sbin;              /* status block IN pointer. */

            DWORD     sbout;             /* status block OUT pointer. */

            DWORD     number_pbs;        /* number of parameter blocks. */

            DWORD     number_sbs;        /* number of status blocks. */

            DWORD     reserved[2];       /* two RESERVED fields. */

            parameter_blockpblist[NPB];/*Parameter Block area. */

            status_block     sblist[NSB};/* Status Block area. */

} command_list;

/* Functions: */

/*_____________________________________________________________________
* put_parameter_block - A function to add a command to the Command List
*____________________________________________________________________*/
void put_parameter_block(pb,clist)
parameter_block pb;
command_list *clist;
```

```
{       /* First: check to see if the list is full */
        if ( (clist→pbin +1) % NPB == clist→pbout)
        {
                /* If the list is full, then handle_full_list */
                /* This may be a function call that waits for an open entry or you could
                choose to return an error. */
                if (!handle_full_list(pb,clist))
                    return;}

        /* Second: put the Parameter Block into the next available location in the list
        */
        clist→pblist[clist→pbin] = pb;

        /* Third: increment the Parameter Block IN index */
        clist→pbin = (clist→pbin + 1) % NPB;

        /* Next issue a Command List Channel Attention */
        /* This is a separate function called to issue a
        Command List Channel Attention. */
        issue_cl_channel_atn();

} /* put_parameter_block */

/
*_____________________________________________________________________
* get_status_block - Read a Status Block returned in the Command List
*____________________________________________________________________*
/
status_block get_status_block(clist)
command_list *clist;
(
        status_block tsb;

        /* First: check to see if a Status Block is available */
        if (clist→sbout == clist→sbin)
        {
                /* If the list is empty there are no blocks to read. Check the adapter
                * Status Port for any error conditions, and return.
                */
                /*This routine could check the Status Port for 11H error code
                (no active command list.) */
                handle_empty_sb_list(clist);
                return(EMPTY);
        }

        /* Second: read the Status Block into a temporary value so that it can be
        returned at the end of the function.
        */
        tsb = clist→sblist[clist→sbout];

        /* Third: increment the Status Block OUT index. */
        clist→sbout = (clist→sbout + 1) % NSB;

        /* Next: return with the Status Block. */
        return (tsb);

} /* get_status_block
```

5

Initiator Mode Pass-through Commands

Introduction

There are two types of commands that you will issue to the RF3880 adapter: Pass-through and adapter Board-control. Board-control commands are covered in Chapter 7. This chapter informs you about Initiator Pass-through commands performed in Initiator Mode. Pass-through commands for Target Mode are described in Chapter 6.

In Initiator Mode, when the RF3880 adapter receives a Initiator Pass-through command Parameter Block, it passes a command or message byte through to the SCSI device that must perform the work. The RF3880 will perform the necessary negotiation and report the completion status.

A Initiator Pass-through command Parameter Block can contain any SCSI command available for the peripheral you are addressing. You should refer to a SCSI and Common Command Set (CCS) specification or your peripheral vendor manual for more information. This chapter describes the Parameter Block structure you should use for Initiator Pass-through commands.

Summary

This chapter summarizes Initiator Pass-through commands and can be used as a reference for:

- Pass-through command Parameter Block format.
- Pass-through command Status Block format.
- Pass-through message Parameter Block format.
- Examples of usage.

Initiator Pass-through Parameter Block Format

In Chapter 4, a Base Parameter Block was introduced and explained in a general context. Parameter Blocks are the same size, with the same general fields, whether you are issuing a Pass-through command or an adapter Board-control command. The information specific to Initiator Pass-through commands will be covered in this section.

Command Format

Use this format when issuing a Parameter Block containing an Initiator Pass-through command:

Address Offset	Byte Memory Address				
	Offset + 0	Offset + 1	Offset + 2	Offset + 3	
00H	Command Identifier				
04H	Flags-2	Flags-1	Addr. Mod.	Target ID	
08H	VME Memory Address				
0CH	Transfer Count				
10H	0 (OP Code)	1	2	3	SCSI 6, 10 or 12 byte Command Descriptor Block
14H	4	5	6	7	
18H	8	9	10	11	

Table 25: Initiator Pass-through Parameter Block

Description of the Fields

The Initiator Pass-through commands Parameter Block fields can be explained as follows:

Command Identifier

The Command Identifier field is used to identify the Status Block associated with a Parameter Block.

It must be a unique value.

Target ID

This value identifies the Target ID of the device that is to receive the command contained in the SCSI Command Descriptor Block.

Address Modifier

This value further defines the VMEbus Address specified for a data transfer.

Note
Valid Address Modifier values can be found by referring to a VME Specification, Revision D.

BITS							
7	6	5	4	3	2	1	0
0	0	Addr Mod					

Table 26: Bit-meanings of Address Modifier Field

Addr Mod Address Modifier - These six bits inform the RF3880 adapter what Address Modifier it should use for data transfer.

Flags-1

The Flags-1 byte controls the command operation. Its bits have these meanings:

Note
Bits 7, 2 and 1 work intra-dependently. They are automatically set by the adapter to specific values for standard SCSI and SCSI-2 command. Their values are described in Table 25 on page 5 - 3.

BITS							
7	6	5	4	3	2	1	0
DBV	0	ICC	0	IRS	DAT	DIR	0

Table 27: Bit-meanings of Flags-1 Field

DIR Direction of Data Transfer - Both the DBV and the DAT bits must be set to 1 for this bit to have meaning for an RF3880 operation. Otherwise its value is ignored.

0 ⇒ Data transfer is from the target.
1 ⇒ Data transfer is to the target.

DAT Data Transfer - This bit is available to be used to enhance efficiency of vendor-unique commands. (It is automatically set for standard commands.) If the command you are issuing is vendor-unique, this bit informs the adapter that a data-in or data-out phase will occur during the command.

0 ⇒ Command has no data phase.
1 ⇒ Data phase will occur.

IRS Inhibit Request Sense - Selects whether Sense Data is to be gathered automatically by the adapter (in response to a Check Condition status from a device) or given to the host to do so.

0 ⇒ Automatic Request Sense issued to device reporting Check Condition.
1 ⇒ Check Condition status of device reported to Host. Host must request the device status.

ICC Inhibit Command Complete Interrupt - Inhibits the interrupt that notifies the host of command completion. By inhibiting interrupts for a number of commands and then enabling an interrupt for one, you can reduce the number of interrupts to service.

0 ⇒ Interrupts enabled for this command.
1 ⇒ Interrupts inhibited for this command.

DBV Data Bits Valid - Use this bit when you wish to override the default values of the DAT and DIR bits to implement vendor-unique commands or alter those automatically set for standard SCSI as shown in Table 25 on page 5 - 3.

0 ⇒ Normal operation.
1 ⇒ DIR and DAT values in the Flag byte are valid and override the defaults.

Setting DBV, DAT and DIR Bits

The DAT and DIR bits, together, indicate whether a data transfer should be expected for a given command, and if so, which direction it will be: to or from the target. The DBV bit gates whether the values for DAT and DIR will be derived from the default table or from the values you assign.

Using the Table Values

The following table of defaults is useful for you to use when you wish to: 1.) alter the default behavior of a standard SCSI command for a specific application or 2.) implement a vendor-unique command.

An example of altering the default function of a command can be demonstrated with the SCSI Format command. If you examine the DAT, and DIR bit values for this command in the table, you see that the adapter assumes:

Opcode	DAT	DIR	Description
04H	1	1	Format Unit

Table 28: Example DAT and DIR bit settings

The ones in the table for DIR and DAT mean that by default the adapter will assume that there is data associated with the command and the direction of data flow is to the target.

If the Format command for your device requires no data transfer, set the Flags-1 byte as follows when you issue the Format command:

DBV = 1, DAT = 0, DIR = 0.

Setting DBV to one tells the adapter to examine your settings for DIR and DAT and their zero settings indicate no data is associated with the command.

An example of the second case, implementing a vendor-unique command, can also be given. Suppose your SCSI device has a vendor-unique command available for Opcode 02H. When you examine the table for the default settings of that opcode you see that the RF3880 adapter assumes:

Opcode	DAT	DIR	Description
02H	0	0	not defined

Table 29: DIR and DAT Settings, Vendor-unique

Note
If you set a combination of DBV, DIR and DAT that is inconsistent with how your target executes the command, you will receive error code 24H, Unexpected SCSI Phase Encountered. You must refer to your vendor-specific documentation for information about your peripheral’s command execution.

In this example, suppose the 02H opcode vendor-unique command involves a data transfer from the device to the adapter. To override the table values and use the opcode for a vendor-unique command set the DBV, DAT and DIR bits of the Flags-1 byte as follows:

DBV = 1; DAT = 1; DIR = 0

This combination means; DBV is set so use Flags-1 field values for DAT and DIR; DAT is set so data will be transferred; DIR is zero so direction of transfer is from the device.

Table of DAT and DIR Automatic Settings

The following table demonstrates the values for the DAT and DIR bits that the RF3880 associates by default with each of the SCSI opcodes:

Opcode	DAT	DIR	Description
00H	0	0	Test Unit Ready
01H	0	0	Rewind/Rezero
02H	0	0	not defined
03H	1	0	Request Sense
04H	1	1	Format Unit
05H	1	0	Read Block Limits
06H	0	0	not defined
07H	1	1	Reassign Blocks
08H	1	0	Read
09H	0	0	not defined
0AH	1	1	Write
0BH	0	0	Seek
0CH	0	0	not defined
0DH	0	0	
0EH	0	0	
0FH	1	0	Read Reverse
10H	0	0	Write Filemarks
11H	0	0	Space
12H	1	0	Inquiry
13H	0	0	Verify
14H	1	0	Recover Buffered Data
15H	1	1	Mode Select
16H	1	1	Reserve
17H	1	1	Release
18H	1	1	Copy
19H	0	0	Erase
1AH	1	0	Mode Sense
1BH	0	0	Start/Stop: Load/Unload
1CH	1	0	Receive Diagnostic
1DH	1	1	Send Diagnostic
1EH	0	0	Prevent/Allow Media Removal
1FH	0	0	not defined
20H	0	0	
21H	0	0	
22H	0	0	
23H	0	0	
24H	0	0	
25H	1	0	Read Capacity
26H	0	0	not defined
27H	0	0	
28H	1	0	Extended Read
29H	0	0	Vendor-unique
2AH	1	1	Extended Write

Table 30: DAT and DIR Automatic Settings

Opcode	DAT	DIR	Description
2BH	0	0	Extended Seek
2CH	1	0	Ciprico Defined
2DH	1	0	
2EH	1	1	Write and Verify
2FH	0	0	Verify
30H	1	1	Search Data Equal
31H	1	1	Search Data High
32H	1	1	Search Data Low
33H	1	1	Set Limits
34H	0	0	Pre-fetch Cache
35H	0	0	Synchronize Cache
36H	0	0	Lock/Unlock Cache
37H	1	0	Read Defect Data
38H	0	0	not defined
39H	1	1	Compare
3AH	1	1	Copy and Verify
3BH	1	1	Write Buffer
3CH	1	0	Read Buffer
3DH	0	0	not defined
3EH	1	0	Read Long
3FH	1	1	Write Long
40H	1	1	Change Definition
41H	1	1	Write Same
42H	0	0	not defined
43H	0	0	
44H	0	0	
45H	0	0	
46H	0	0	
47H	0	0	
48H	0	0	
49H	0	0	
4AH	0	0	
4BH	0	0	
4CH	1	1	Log Select
4DH	1	0	Log Sense
4EH	0	0	not defined
4FH	0	0	
50H	0	0	
51H	0	0	
52H	0	0	
53H	0	0	
54H	0	0	
55H	1	1	Mode Select (10)
56H	0	0	not defined
57H	0	0	
58H	0	0	

Table 30: DAT and DIR Automatic Settings

Opcode	DAT	DIR	Description
59H	0	0	not defined
5AH	1	0	Mode Sense (10)
5BH	0	0	not defined
5CH	0	0	
5DH	0	0	
5EH	0	0	
5FH	0	0	
60H thru A3H	0	0	not defined
A4H	1	0	Ciprico defined
A5H	0	0	Move Medium/ Play Audio
A6H	0	0	Exchange Medium
A7H	0	0	not defined
A8H	1	0	Read (12)/Get Message
A9H	0	0	Play Track Relative
AAH	1	1	Write (12)/Send Message
ABH	0	0	not defined
ACH	0	0	Erase (12)
ADH	0	0	not defined
AEH	1	1	Write and Verify (12)
AFH	0	0	Verify (12)
B0H	1	1	Search Data High (12)
B1H	1	1	Search Data Equal (12)
B2H	1	1	Search Data Low (12)
B3H	1	1	Set Limits
B4H	0	0	not defined
B5H	1	0	Request Volume Element Address
B6H	1	1	Send Volume Tag
B7H	1	0	Read Defect Data (12)
B8H	1	0	Read Element Status
B9H thru BFH	0	0	not defined
C0H	0	0	Ciprico defined
C1H	0	0	not defined
C2H	1	0	Ciprico defined
C3H	1	0	Ciprico defined
C4H thru CEH	0	0	not defined
CFH	0	0	Ciprico defined
D0H thru D5H	0	0	not defined
D6H	0	0	Ciprico defined
D7H	0	0	Ciprico defined
D8H thru E3H	0	0	not defined

Table 30: DAT and DIR Automatic Settings

Opcode	DAT	DIR	Description
E4H	1	0	Ciprico defined
E5H thru EFH	0	0	not defined
F0H	0	0	Ciprico defined
F1H	0	0	not defined
F2H	0	0	not defined
F3H	0	0	not defined
F4H	0	0	Ciprico defined
F5H thru FFH	0	0	not defined

Table 30: DAT and DIR Automatic Settings

Flags-2

The bits of this field are used to select SCSI-2 and additional command control features implemented by the RF3880 adapter. The meanings of the bits are as follows:

BITS							
7	6	5	4	3	2	1	0
0	0	0	0	JSM	IAD	TAG OPTS	

Table 31: Bit-meanings of Flag-2 Field

TAG OPTS - These two bits let you choose the tagged queuing method you wish used with the current command. You can choose one of three types of messages for the RF3880 adapter to use when issuing the command.

Note
Depending on the SCSI-2 device you are using, you may also need to enable tagged queuing with a MODE SELECT page 0AH, Common Device-Type Control Parameters command to the device. See your vendor-supplied peripheral documentation for more information.

TAG OPT Value	Meaning:
00	SIMPLE QUEUE TAG MESSAGE: The command may be processed by the SCSI-2 device out of sequence for efficiency. Example: device may sort commands by logical block address.
01	ORDERED QUEUE TAG MESSAGE: The command must be executed by the SCSI-2 device in the order sent by the adapter (in order with respect to other commands sent with the ORDERED QUEUE TAG MESSAGE).
10	HEAD OF QUEUE TAG MESSAGE: The command is executed by the SCSI-2 device before all other commands previously sent, including other Head of Queue Tag commands but excluding command currently in process.
11	Reserved

Table 32: Meaning of TAG OPTS Bits

In order for these bits to apply to the current operation you must have already enabled tagged queuing with either the Unit Options or Extended Unit Options command. These commands are covered in detail in Chapter 7, Board-control Commands. If you do not enable tagged queuing with the Unit Options or Extended Unit Options command, the TAG OPTS bits are ignored.

For more information about the use of tagged queuing, see the usage discussion in Chapter 8.

IAD Inhibit Automatic Disconnect - If you enable SCSI disconnect privilege with the DIS bit of the General Options Board-control command (Bit 0 of Select Flags field), disconnects will automatically be tried for any command. Use the IAD bit to disable SCSI device disconnect/reconnect privilege for the current command. This can be important for performance reasons.

0 ⇒ Determine disconnect from the DIS bit of General Options command.
1 ⇒ Ignore DIS bit, inhibit the disconnect privilege.

JSM Just Send Message - Use this bit to change the Initiator Pass-through Command Parameter Block to a Pass-through Message Parameter Block. In this Message mode, the adapter selects a device and only sends a message. This provides a measure of error recovery at the host level.

0 ⇒ Pass-through command operation as usual.
1 ⇒ Use the Pass-through Message Parameter Block format.

This message-only format is illustrated on page 5 - 11, following the description of the Pass-through command structure.

VME Memory Address

When the SCSI operation involves a data transfer, this field contains the beginning address for the transfer. Note: transfers must be word-aligned.

If the SCSI command you are issuing does not require a data transfer, the adapter ignores this field and the Address Modifier field.

Transfer Count

This field determines the number of bytes that will be transferred to or from host memory. For some SCSI commands this is an indeterminate number. In that case, there are two ways to use this field:

1. You can place a value in the field to represent the maximum number of bytes the adapter may transfer. Then, if the target attempts to transfer more data than this, the adapter will discontinue the operation and return a Status Block.

2. Alternately, you could place a zero value in this field. This causes the adapter to continue to accept data transfer requests from the target until command is complete.

SCSI Command Descriptor Block

These 12 bytes of the Initiator Pass-through Parameter Block are reserved for the actual SCSI command that will be passed to the drive. Except for the first byte, which contains the command opcode, each SCSI command uses the bytes for a different purpose; see a SCSI specification or vendor-supplied peripheral documentation for details about the commands and fields.

SCSI commands may be 6, 10, or 12 bytes in length. Any of the 12 bytes reserved for the SCSI command that are unused are ignored. The SCSI command block is passed to the SCSI device exactly as presented. The RF3880 adapter doesn't alter any of the fields you set up.

Message Format

When you set the JSM bit in the Flags-2 field of the Pass-through Parameter Block, the format of the Parameter Block changes:

Address Offset	Byte Memory Address			
	Offset +0	Offset +1	Offset +2	Offset +3
00H	Command Identifier			
04H	Flags 2	Flags 1	Addr. Mod.	Target ID
08H	VME Memory Address			
0CH	Transfer Count			
10H	Message Code	LUN	2	3
14H	4	5	6	7
18H	8	9	10	11

Table 33: Pass-through Message Parameter Block

Description of the Fields

All fields that appear in light text above are ignored when the JSM bit of the Flags-2 byte is set.

The Command Identifier and Target ID fields work the same as explained in the preceding pages. The JSM bit of the Flags-2 field is the only bit in that field that is valid.

Message Code

Currently there are three messages that are valid to appear in this field. They are:

06H - ABORT
0CH - BUS DEVICE RESET
0EH - CLEAR QUEUE

Other messages will not be passed to the device and will create an error status. Error code returned is 01H (invalid command).

LUN

This field has the same format as specified in the SCSI specification for LUN (Logical Unit Number); the 3 most significant bits hold the LUN value.

Initiator Pass-through Status Block Format

In Chapter 4, a Base Status Block was introduced and explained in general terms. The Status Block for an Initiator Pass-through command encompasses the same fields used for the Board-control commands, plus it holds information that is particular to a SCSI transaction: a SCSI status byte, and eight bytes of Sense data returned from the SCSI device.

The Sense Data fields are an important part of the Status Block. Sense data is acquired directly from the SCSI device; you can enable the adapter to automatically request Sense Data in response to an error condition reported from the device. What you will see in these fields is affected by your use of the Unit Options or Extended Unit Options Board-control commands. If you use one of these commands to have more than eight bytes of Sense Data returned, you will receive multiple Status Blocks.

This section includes information about all the fields you will see when examining a Status Block returned from a Initiator Pass-through command.

Status Block Format

This is the format you can expect from an Initiator Pass-through Status Block:

Note
The fields displayed in 08H and 0CH are the default Sense Bytes returned by Initiator Pass-through

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Reserved	SCSI Status	Error	Flags
08H	0 = Class/Code	1 = Segment	2 = SCSI Flags	3 = Info Bytes
0CH	4 = Info Byte	5 = Info Bytes	6 = Info Byte	7 = Ex Length

Table 34: SCSI Pass-through Status Block

Description of Fields

Following is a brief description of each of the fields:

Command Identifier

This value links a Status Block with a Parameter Block.

Flags

This byte can be polled to ascertain whether, and in what condition, a command completed:

BITS							
7	6	5	4	3	2	1	0
CC	ERR	RTY	DTT	DTG	CSB	0	TMS = 0

Table 35: Bit-meanings of Flags Field

TMS Target Mode Status - This bit is will not be true for an Initiator Pass-through command status block.

CSB Continued Status Block - This bit indicates that the current Status Block is one of several Status Blocks generated because extended Sense Bytes are enabled. Only valid for Pass-through commands issued through Command List.

0 ⇒ Status Block is first (or only) Status Block
1 ⇒ Status Block is one of a series of Status Blocks.

DTG Data Transfer Greater - Indicates when an Initiator Pass-through command completes with more bytes requested by the SCSI device than was specified in the Transfer Count field.

0 ⇒ Data transfer was not greater.
1 ⇒ Data transfer was greater.

DTT Data Transfer Truncated - Indicates when an Initiator Pass-through command completes with fewer bytes transferred than requested in the Transfer Count field.

0 ⇒ Data transfer was not truncated.
1 ⇒ Data transfer was truncated.

RTY Retry - Indicates whether an Initiator Pass-through command required retries to complete.

0 ⇒ No retries were required.
1 ⇒ Retries were required to complete.

ERR Error Status - Identifies when a command, Board-control or Pass-through, completes with an error.

0 ⇒ No error occurred.
1 ⇒ An error occurred.

CC Command Complete - Indicates whether a command has finished.

0 ⇒ Command not complete.
1 ⇒ Command complete.

Error

When an adapter-detected error occurs, this byte contains an error code to describe the condition. See Appendix A.

SCSI Status

When a SCSI device completes a command, during the STATUS phase of the SCSI bus it returns a SCSI Status Byte Code to the adapter. The adapter reports that value in this field. The field has these bit meanings:

BITS							
7	6	5	4	3	2	1	0
RES	RES	Status Byte Code					RES

Table 36: Bit-meanings of SCSI Status Field

RES - SCSI reserved.

Status Byte Code - Implementation of these bits is required for all devices that adhere to SCSI specifications. See the table below for information about what they mean.

Further information about the Status Byte Code can be found in the SCSI, SCSI-2, or SCSI-3 specification.

Status	HEX Code*	7	6	5	4	3	2	1	0
Good	00H	R	R	0	0	0	0	0	R
Check Condition	02H	R	R	0	0	0	0	1	R
Condition Met/Good	04H	R	R	0	0	0	1	0	R
Busy	08H	R	R	0	0	1	0	0	R
Intermediate/Good	10H	R	R	0	1	0	0	0	R
Intermediate/Condition Met/Good	14H	R	R	0	1	0	1	0	R
Reservation Conflict	18H	R	R	0	1	1	0	0	R
Queue Full	28H	R	R	1	0	1	0	0	R

Table 37: Meaning of Status Byte Code

SCSI Status Field during Unexpected SCSI Phase

In the event that the adapter encounters an unexpected SCSI phase it does two things: posts an error (24H) in the Error field of the Status Block, and reports the phase encountered in the last three bits of the SCSI Status field. The bits correspond to the MSG, C/D, and I/O signals. At the time of an Unexpected SCSI Phase Encountered error, the SCSI Status field has the following bit-meaning:

BITS							
7	6	5	4	3	2	1	0
X	X	X	X	X	M	C	I

Table 38:

I - I/O (Input/Output) Signal from Target

C - C/D (Control/Data) Signal from Target

M - MSG (Message) Signal from Target

X - May be either 0 or 1

The unexpected phase encountered can be determined by interpreting the state of the three signals:. This is described below:

Signal			Phase	Direction of Transfer
MSG	C/D	I/O		
0	0	0	Data Out	Initiator to Target
0	0	1	Data In	Initiator from Target
0	1	0	Command	Initiator to Target
0	1	1	Status	Initiator from Target
1	0	0	Reserved	—
1	0	1	Reserved	—
1	1	0	Message Out	Initiator to Target
1	1	1	Message In	Initiator from Target

Table 39: Meaning of Status Byte - Unexpected Phase

See your SCSI or SCSI-2 specification for more information about SCSI phases.

Sense Bytes - When enabled to do so (the IRS bit of the Flags-1 field of the Pass-through command is not set), the RF3880 adapter automatically responds to Check Condition status from a device, with a Request Sense command.

The device answers the Request Sense command by returning information about its condition. This information is called Sense Bytes. Depending on the peripheral, up to 256 Sense Bytes can be returned in response to the Request Sense command.

There are three ways that Sense Bytes can be reported to you by the RF3880 via the Sense Bytes fields of the Status Block:

- The first eight bytes of Sense data returned (this is the default).
- Up to 32 of the first sequential Sense Bytes.
- Up to 16 of any of the 256 possible Sense Bytes returned.

You can specify one of these methods, for Sense Bytes to be reported in the Status Block of Initiator Pass-through commands, by using the Unit Options or Extended Unit Options Board-control command (See Chapter 7).

If no Check Condition status occurred, or the automatic Request Sense capability of the adapter is not enabled, the Sense Bytes fields will all be zero. See page 5 - 4 for more information about how to inhibit the automatic Request Sense command (using the IRS bit).

Default Sense Bytes

You will receive the default eight bytes of Sense data if you place a zero in the Sense Count field and zeroes in all fields of the Selected Sense Bytes of the Unit Options or Extended Unit Options Board-control command.

The default values returned are:

Byte Memory Address			
Offset + 0	Offset + 1	Offset + 2	Offset + 3
Command Identifier			
Reserved	SCSI Status	Error	Flags
0 = Class/Code	1 = Segment	2 = SCSI Flags	3 = Info Byte
4 = Info Byte	5 = Info Byte	6 = Info Byte	7 = Ex Length

Table 40: Default Ordering of Sense Bytes

Only one Status Block is required to return the default Sense Bytes.

Up to 32 Sequential Sense Bytes

This option is chosen by placing a number (up to 32) in the Sense Count field of either the Unit Options or Extended Unit Options command. The Selected Sense Bytes fields are ignored.

This is useful if all pertinent information occurs within the first 32 bytes of Sense data returned from your device.

If you enable more than eight bytes of sequential Sense data to be returned, they will be returned in multiple Status Blocks. For example, the first sixteen would appear in this order:

Byte Memory Offset				
Offset + 0	Offset + 1	Offset + 2	Offset + 3	
Command Identifier				
Reserved	SCSI Status	Error	Flags	
0 = Class/Code	1 = Segment	2 = SCSI Flags	3 = Info Byte	Sense Byt in Status Block #1
4 = Info Byte	5 = Info Byte	6 = Info Byte	7 = 8H	
Command Identifier				
Reserved	SCSI Status	Error	Flags	
8	9	10	11	Sense Byt in Status Block # 2
12	13	14	15	

Table 41: Ordering of 1st 16 Sequential Sense Bytes

Selectable Sense Bytes

You also have the option to select which (up to 16) of the 256 possible Sense bytes be returned in the Status Block and in what order they should appear.

- The Unit Options command allows you to choose 8 selectable Sense Bytes.
- The Extended Unit Options command allows you to choose 16 selectable Sense Bytes. You will receive multiple Status Blocks.

You can specify Selectable Sense Bytes by placing a zero in the Sense Count field and filling each of the Selected Sense fields with the Sense Byte number you wished returned in that field.

For example, the Sense Bytes fields of a Status Block could return the following Sense Bytes in the following order if the Extended Unit Options command had been used.

Byte Memory Offset				
Offset + 0	Offset + 1	Offset + 2	Offset + 3	
Command Identifier				
Reserved	SCSI Status	Error	Flags	
0	1	2	3	Sense Byt in Status Block #1
4	5	6	7	
Command Identifier				
Reserved	SCSI Status	Error	Flags	
56	57	12	19	Sense Byt in Status Block #2
122	33	12	23	

Table 42: Example of Selected Sense Bytes

Status Block #1 contains the first eight bytes with the Info Bytes arranged from LSB to MSB, and Status Block #2 returns Sense Bytes that are selected from among the other 248 Sense Bytes, based on information in the vendor-supplied peripheral documentation.

See Chapter 7 for more information about the Unit Options and Extended Unit Options commands and how to set-up the Status Block Sense bytes the way you wish.

6

Target Mode Pass-through Commands

Introduction

With the RF3880 you have the option to use the adapter as a target on the SCSI bus. The RF3880 may operate as a target, as an initiator, or as both a target and an initiator at the same time.

When in Target Mode, the RF3880 can be used to emulate any of the defined SCSI peripheral device types.

As an initiator the RF3880 receives two types of commands: Board-level and Pass-through. This is also true when it is operating as a target.

Summary

This chapter explains how to setup the Target Mode feature of the RF3880 and describes the following aspects of its use:

- Command and status implementation.
- Command operation.
- Sequence of operation.

This chapter is useful to use to understand the Target Mode of the RF3880.

Overview of Target Mode

Of course, if you are familiar with the SCSI specification, you already know that operation over the SCSI bus takes place between two parties: the initiator of the operation, and the target of the operation. In the SCSI specification, protocol for each party is exactly detailed. A target has certain responsibilities in a transaction, and an initiator has other specific obligations.

Most of this manual has been devoted to describing how the Rimfire 3880 can be used as an initiator on the SCSI bus. This chapter explains how the RF3880 can be used as a SCSI target. It explains a mode of operation for the adapter that is called Target Mode.

What is Target Mode?

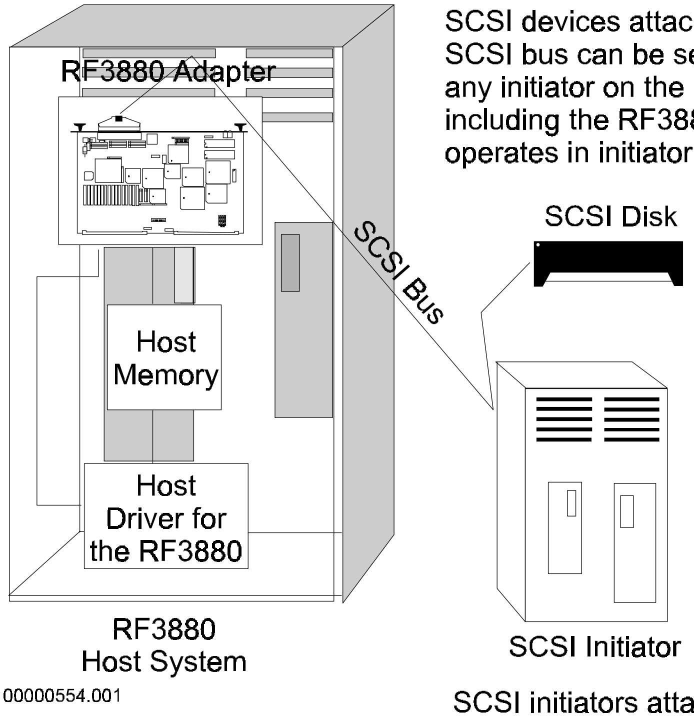
The Ciprico implementation of Target Mode on the RF3880 offers flexibility to the person writing a driver for it. The adapter actually makes very few independent decisions. Instead, when acting as a target, transaction information is passed to the Host, and the Host frames the response that the adapter will make as the target. The driver that handles Host business must be written to follow the SCSI specification as well as the Ciprico protocol presented in this chapter.

When Target Mode is enabled, the adapter will accept selections from initiators and initiate reselections. It will also continue to function as an initiator, selecting targets at the request of the Host, and accepting reselections.

The RF3880 handles basic SCSI Hardware protocol, but passes thorough command decisions to the Host driver for the RF3880 when in Target Mode.

A data structure that communicates selection data is put into memory by the RF3880 and a command that communicates response data is sent by the Host driver.

The Host driver for the RF3880 must make decisions about how the RF3880 should respond to selection by any initiator when in Target Mode, and process the status passed back from the adapter.



SCSI devices attached to the SCSI bus can be selected by any initiator on the bus, including the RF3880, when it operates in initiator mode.

SCSI initiators attached to the same SCSI bus as the RF3880 are free to negotiate with the RF3880 as if it were a SCSI peripheral device.

Figure 18 RF3880 in Target Mode

What must the adapter accomplish in Target Mode?

There are special commands, setup rules and status involved with Target Mode that are necessary to accomplish the work required of the adapter:

The adapter must provide a means to inform the Host that it has been selected by an initiator.

Few responses of the adapter are automatic—most often it is the responsibility of the Host to determine action after a selection of the RF3880 occurs over the SCSI bus. Communication of a selection by an initiator is done via a data structure created in memory (Target Selection Data Structure). The address of the memory location that the adapter is to use, is part of the *Enable Target Mode* command. As soon as it is selected, the adapter writes information about the selection into the appropriate memory and interrupts the Host.

For more information about the *Target Mode Enable* command, see Chapter 7, Board-control Commands.

For more information about the Target Selection Data Structure, see page 6 - 10.

The adapter must have a means for the Host to communicate appropriate responses for the adapter to use.

By use of a command similar to the Initiator Pass-through commands, the Host is able to tailor the type of response that the RF3880 (as target) will return. The command is called *Target Mode Response* command. When this command is received, the adapter reads and uses the data specified in the command Parameter Block to respond to the initiator that selected it.

For more information about the *Target Mode Response* command, refer to page 6 - 15.

The adapter must have a means of communicating status back to the Host.

As it does as an initiator, the RF3880 operating in Target Mode returns a Status Block to the Host when an operation completes. Depending on the command and how it completed, the Status Block returned may contain additional information from the Initiator.

For more information about the Status Blocks returned in Target Mode, refer to page 6 - 19.

The adapter must have a means of handling SCSI protocol issues.

The SCSI protocol is flexible in allowing a variety of negotiations to occur between initiator and target. Tagged Queuing, wide and/or synchronous transfers, response to other messages, and transmission errors all must be handled according to the specification to allow for compatibility. The RF3880 passes message decisions to the Host for processing except: Initiator-detected Error messages, and Message Parity error messages. The adapter handles wide or synchronous transfer negotiation with an initiator and adapter-detected parity errors without interaction with the Host.

For more information about the adapter handling of SCSI protocol details, see Chapter 8, Details of Usage.

Overview of Special Target Mode Commands

To implement the Target Mode described in the previous pages, three new commands and a new type of Status Block were used. These are described briefly next so the commands will be familiar as the Sequence of Operation is described.

New commands

- *Target Mode Enable* command
- *Target Mode Disable* command
- *Target Mode Response* command

Note
For a complete description of the *Target Mode Enable* command and the *Target Mode Disable* command, see Chapter 7. The *Target Mode Response* command is described fully later in this chapter.

Target Mode Enable Command

When you want the adapter to begin operating in Target Mode, you must issue a *Target Mode Enable* command to the adapter. This command must be issued after the host has set up the memory for the Target Selection Data Structure that the adapter will use. Until this command is issued, the adapter will respond to all selection attempts by disconnecting from the SCSI bus.

Target Mode Disable Command

The *Target Mode Disable* command is issued when you wish to stop receiving selections from initiators on the SCSI bus.

Target Mode Response Command

When ready with the response for the initiator that has selected the RF3880, use the *Target Mode Response* command to inform the adapter what action to perform. *Target Mode Response* commands are queued in the adapter for execution in the same way as initiator commands.

Command Descriptor Block Group Codes

The RF3880 will decode the group number of a SCSI command, and only receive the number of command bytes specified by that group code. If more information is required by a vendor-specific command, the additional information must be sent in the Data phase.

The RF3880 makes the following assumptions about the length of a Command Descriptor Block received from the Host:

Note
* During Target Mode operations, the Tag bit in the *Unit Option* (08H) command's Unit Flag byte controls whether Group 2's 6- or 10-byte commands are accepted. This applies to the *Extended Unit Options* (18H) command as well, see Chapter 7 for further details.

Group	OP Codes	Description
0	00 - 1F	6-byte commands
1	20 - 3F	10-byte commands
2*	40 - 5F	6-bytes (no SCSI-2 devices configured) 10 bytes (SCSI-2 devices configured)
3-4	60 - 9F	6-byte commands (reserved)
5	A0 - BF	12-byte commands
6	C0 - DF	6-byte commands (vendor specific)
7	E0 - FF	10-byte commands (vendor specific)

Table 43: SCSI Command Descriptor Block Groups

Command Operation

In this section, the sequence of events that can be expected and the parts that the adapter and Host must play are described.

Sequence of Events—Enabling Target Mode

The following sequence must occur to ensure proper operation of Target Mode:.

1. Host allocates 32 bytes of memory for the Target Selection Data Structure, to be used by the adapter when it is selected by an initiator.

Note
When Target Mode is terminated, whether by a *Disable Target Mode* command or by a SCSI bus Reset, the Target Selection Data Structure will be written by the adapter, with the TME bit of the Flags field cleared. Additionally, if a SCSI bus reset occurred, an error of 27H will be reported in the Error field

2. The Host issues a *Target Mode Enable* command to the RF3880. The address of the Target Selection Data Structure is passed in the command.
3. The adapter is ready to accept selection by an initiator and will remain so until a SCSI Reset occurs or the Host issues a *Target Mode Disable* command.

Sequence of Events—Receiving a command from an Initiator

The following sequence occurs when an initiator selects the RF3880:

1. An initiator selects the RF3880 as a SCSI target device, and sends a SCSI command.
2. Using the command data it receives from the initiator, the adapter builds a Target Selection Data Structure in memory, at the address indicated in the *Target Mode Enable* command.
3. The adapter sets the SAV bit in the Semaphore field of the Target Selection Data Structure to indicate that a selection has occurred.
4. The adapter disconnects from the SCSI bus. (The adapter will not disconnect from the SCSI bus if disconnect privilege is not granted in the Identify message received from the initiator.)
5. The adapter interrupts the Host. (The adapter will not interrupt if a Selection Interrupt Level of zero in the *Target Mode Enable* command is used.)

Note
Until the Host responds to selection (by writing a 2 to the Channel Attention Port), the adapter will queue other selections. Once the adapter's queue is full (approximately 60 selections) the adapter will respond to any further selection attempts by sending a Busy status to the initiator(s) trying to select it

Any errors that are non-recoverable will be indicated in the Target Selection Data Structure.

6. The Host must read the Target Selection Data Structure. The Host should clear the SAV bit in the Semaphore byte of the data structure after it is read.
7. The Host must write a 2 to the Channel Attention Port of the adapter to indicate the data has been read.
8. The adapter is ready to send another selection area to the Host once it is selected again by an initiator.

Sequence of Events—Responding to an Initiator

The following events must occur for the adapter to provide a response to an initiator (it is assumed that a command to start Command List operation has already been issued to the adapter):

1. The Host interprets the Target Selection Data Structure.
2. The Host writes Target Response data into a *Target Mode Response* command Parameter Block.
3. The Host issues the command to the RF3880. This command must be issued from a Command List.
4. The RF3880 reselects the initiator specified in the Target Response Data Structure, and performs the work requested.
5. The adapter sends the SCSI status byte and Command Complete message as specified by the *Target Mode Response* command.
6. The adapter returns the SCSI bus to the Bus Free phase.
7. The adapter sets completion status for the *Target Mode Response* command and sends a Status Block to the Host.
8. The Host receives completion status for the *Target Mode Response* command and determines if any further action is required.

Note
Instead of sending the status and Command Complete message at step 7, the host may only have part of the data required to be transferred. If so, the Host can place Disconnect and Save Data Pointer messages in this *Target Mode Response* command and follow with a second *Target Mode Response* command containing the remaining data and the status byte and Command Complete message.

Other options are also available, see the description of the Response Control Flags field in the *Target Mode Response* command on page 6 - 17.

Target Selection Data Structure

The Target Selection Data Structure is used by the adapter to give the Host driver information about the transaction requested of the adapter by a SCSI initiator.

The address of the area reserved by the Host to accommodate the Target Selection Data Structure, is passed to the RF3880 in the *Target Mode Enable* command (described in Chapter 7). The structure has the following format:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	SCSI flags	Reserved	Error	Flags
08H	Msg Byte Rcd	CDB Length	Init ID	ID Msg Received
0CH	Queue Tag Msg	Queue Tag ID	Reserved	
10H	SCSI CDB Received			
14H	SCSI CDB Received			
18H	SCSI CDB Received			
1CH	Reserved			Semaphore

Table 44: Target Selection Data Structure

To buffer Host operations from Target Mode operations, information destined for the Target Selection Data Structure is queued by the RF3880, as initiator selections occur on the SCSI bus. Information from about sixty selections from initiators can be queued at a time. See Chapter 8 for more information about the Selection Queue.

Command Identifier

The Command Identifier that labels the *Target Mode Enable* command is copied by the adapter into the Target Selection Data Structure when a selection occurs. The Host driver may use this for verification of the source of the selection data.

Flags Field

This field contains flags that report the status of the selection process. The format of this field is as follows:

BITS							
7	6	5	4	3	2	1	0
CC	ERR	RTY	0	0	0	0	TME

Table 45: Meaning of Flags Field

TME Target Mode Enabled - This bit indicates the current status of Target Mode.

0 ⇒ Target Mode has been terminated. This Target Selection Data Structure was not due to an initiator selecting the adapter. (Either Target Mode was disabled by a SCSI Reset, or by the *Target Mode Disable* command.)
1 ⇒ Target Mode is enabled. This Target Selection Data Structure was sent due to an initiator selecting the adapter.

RTY Retry - Indicates whether a selection process required retries to complete.

0 ⇒ No retries were required.
1 ⇒ Retries were required to complete.

ERR Error Status - Identifies when a selection process completes with an error. Used in conjunction with the Error field, indicates if error was recoverable.

0 ⇒ No error occurred, or if non-zero value exists in the Error field, error was recoverable.
1 ⇒ An error occurred. Also set if Target Mode was terminated due to a SCSI Reset.

CC Command Complete - Indicates whether a selection process has finished.

0 ⇒ Command not complete.
1 ⇒ Command complete.

Error

When the adapter detects an error, this byte contains an error code to describe the condition. There are special Target Mode error codes that indicate Target Mode error conditions. These codes are listed along with the other adapter error codes in Appendix A.

SCSI Flags

This byte indicates the status of the current command, and of the SCSI bus. You must use it to check for certain conditions that may have occurred and need specific action. The format is as follows:

BITS							
7	6	5	4	3	2	1	0
Error Sequence Code				CMD	MSG	CHK	DIS

Table 46: Meaning of SCSI Flags Field

DIS Disconnect - The bit indicates whether the adapter has disconnected from the SCSI bus while processing this command.

0 ⇒ The adapter has not disconnected.
1 ⇒ The adapter has disconnected to process this command.

CHK Check Condition - Because of an error condition, the RF3880 may attempt to send a Check Condition status followed by a Command Complete message to the initiator.

0 ⇒ No Check Condition status was sent.
1 ⇒ A Check Condition status and Command Compete message to the initiator were attempted.

MSG Message - The adapter (as Target) may receive a Message from the initiator. This bit indicates whether a Message was received and can be found in the Message Byte Received Field.

0 ⇒ No Message Byte was received.
1 ⇒ The Message Byte Received field contains a valid Message from the initiator.

CMD Command Bytes Received - This bit indicates whether the adapter has placed SCSI Command Bytes into the CDB Received area of the Target Selection Data Structure.

0 ⇒ No SCSI commands were received.
1 ⇒ A SCSI command was received. The CDB Received area is valid, and CDB Length field is valid.

Error Sequence Code - The Error Sequence Code tells you, according to the following table, where in the selection process an unrecoverable error occurred, or a Message Byte was sent from the initiator. Some of these codes are not possible due to a selection, but may be reported in a *Target Mode Response* command Status Block.

Sequence	Description
0	No message was received.
1	Message received after the RF3880 sent the Status or First Message Byte.
2	Message received after sending the Message Byte.
3	Message received after sending the Queue Tag Message.
4	Message received after a Data Transfer phase.
5	Message received after Selection phase.
6	Message received after sending the Identify message.
7	Message received after a Command phase.
8	Message received in response to SDTR message.
9	Message received in response to WDTR message.

Table 47: Target Mode Error Sequence Code

Identify Message Received

This byte contains the last Identify message that was received from the initiator. This byte includes the logical unit number (LUN) of the target being selected. If the Error Sequence Code is 5, or the Error field is 33H, this field is invalid.

Initiator Identifier

This byte contains the SCSI ID of the initiator that selected the adapter. If the Error field is 33H, this field is invalid.

Command Descriptor Block Length

This byte indicates the number of command bytes that were received from the initiator, and have been passed on in the CDB fields of the Target Selection Data Structure. If the CMD bit is clear (in the SCSI Flags field), this field is invalid.

Message Bytes Received

This byte contains a SCSI message byte received from the initiator. If the MSG bit is clear (in the SCSI Flags field), this field is invalid. Also, the value of the Error Sequence Code must be considered when examining the Message Byte.

Queue Tag Identifier

This byte contains the queue tag identifier assigned to this SCSI command by the initiator. If the Queue Tag message is 0H, this field is invalid (no queue tag was received).

Queue Tag Message

This byte contains any Queue Tag message received from the initiator.

SCSI CDB fields

These 12 bytes contain the SCSI 6, 10, or 12 byte command received from the initiator that selected the RF3880. The number of bytes that are valid is specified in the CDB Length field of the Target Selection Data Structure. If the CMD bit is clear (in the SCSI Flags field), this field is invalid.

Semaphore Field

Both the Host driver and the adapter use this field to flag that an action has been taken. The format is as follows:

BITS							
7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	SAV

Table 48:

SAV Selection Area Valid - After the adapter has been selected by an initiator and has placed valid data in the Target Selection Data Structure, it sets the SAV bit to inform the Host. This bit is set immediately before the adapter issues a Selection Interrupt (if not prohibited by a zero Interrupt level in the *Target Mode Enable* command).After the data structure is read, it is up to the Host to zero the structure. The RF3880 will not write new data to this area until it has received a Channel Attention write of 2.

0 ⇒ The Target Selection Data Structure has been read by the Host and the area is ready for use by the adapter again.
1 ⇒ The RF3880 has placed valid data into the memory location to be read by the Host.

Target Mode Response Command

After the adapter has received a selection from an initiator and the Host has processed the data that it wrote into the Target Selection Data Structure, the Host must issue a *Target Mode Response* command to communicate information to be used by the RF3880 to respond to the initiator that selected it.

Note
This command must be issued from a Command List in order to accommodate the possibility of two Status Blocks being returned.

Target Mode Response commands are queued by the adapter for execution in the same manner in which the adapter queues commands when it is operating as an initiator. Multiple operations in a sequence can be performed on the SCSI bus. An example: first *Target Mode Response* command operations—send data, send Save Data Pointer message, send Disconnect message, and disconnect; second *Target Mode Response* command operations—reconnect, send more data, send status, send Command Complete message, and disconnect.

An initiator can send an unexpected message at any time. The adapter will always accept the message, writing it into the Status Block in the Message Byte Received field.

Command Format

The Parameter Block format for the *Target Mode Response* command looks like this:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	LUN	Flags-1	Addr Mod	Initiator ID
08H	VME Memory Address			
0CH	Transfer Count			
10H	Reserved		Rspse Cntrl Flgs	Reserved
14H	Queue Tag Msg	Queue Tag ID	Status/1st Msg	Message Byte
18H	Reserved			

Table 49: Target Mode Response Parameter Block

Description of Fields

Each of the fields used by the *Target Mode Response* command are explained in the following section:

Command Identifier

The Command Identifier is used to identify the Status Block associated with a Parameter Block. It must be a unique value.

Initiator ID

This field must contain the SCSI ID of the initiator that is to be reselected.

Address Modifier

This is the value that the adapter is to drive onto the VMEbus Address Modifier lines when selecting the memory address for the data transfer.

Flags-1

The Flags-1 byte contains specific controls for the operation of the *Target Mode Response* command. The format is as follows:

BITS							
7	6	5	4	3	2	1	0
0	0	ICC	TMO	0	DAT	DIR	0

Table 50: Bit-meanings of Flags-1 Field

DIR Direction of Data Transfer - If a data transfer is required to complete a command, use this bit to inform the adapter of the direction the data is to be transferred. This bit is only significant if the DAT bit is set to one.

Note
The RF3880 is acting as a target when executing a *Target Mode Response* command.

0 ⇒ Data transfer is from the initiator.
1 ⇒ Data transfer is to the initiator.

DAT Data Transfer - Use this bit to inform the adapter that the command it is to perform involves a data transfer. This bit is used by the adapter in conjunction with the DIR bit.

0 ⇒ Command requires no Data phase.
1 ⇒ A Data phase is required for this command.

TMO Target Mode Operation - This bit distinguishes Target Mode Response commands from Initiator Mode Pass-through commands

0 ⇒ Initiator Mode Pass-through command
1 ⇒ *Target Mode Response* command.

ICC Inhibit Command Complete Interrupt - This bit is used to inhibit the interrupt that would normally notify you that the command has completed.

0 ⇒ Interrupt using the Interrupt level indicated in the Interrupt field of the *Start Command List* command, when the command has completed.
1 ⇒ Do not interrupt to indicate command completion.

LUN

When the initiator selected the RF3880, it specified a LUN in its Identify message. For reselection, this byte contains that same logical unit number (LUN).

VME Memory Address

When the SCSI operation requested by the initiator that selected the RF3880 involves a data transfer, this field contains the beginning Host memory address for the transfer. This field is ignored for SCSI transactions that require no transfer of data.

Transfer Count

Note
A value of zero indicates a transfer of 4 billion bytes, and should not be used.

The value placed in this field indicates the number of bytes to be transferred to or from Host memory. For some SCSI commands this is an indeterminate number. In that case, use the number of bytes allocated in Host memory for this Data area.

Response Control Flags

Since the adapter, acting as a Target, controls the phases of the SCSI bus, this field is important to use as a means of informing the RF3880 of what actions it should take on the SCSI bus, during which phase. The format is as follows:

BITS							
7	6	5	4	3	2	1	0
SMB	LSC	DSC	NSC	CMD	BFR	DSD	CSC

Table 51: Meaning of Response Control Flags Field

CSC Connected SCSI Command - The following conditions will cause the adapter to hold connection to the SCSI bus: disconnects are not allowed by either initiator or Host, ATN was asserted by the initiator, or certain error conditions occurred. To continue processing the nexus for which the bus is held, the Host must send a *Target Mode Response* command that has this bit set. This *Target Mode Response* command's Initiator ID, LUN, and Queue Tag ID fields must exactly match the nexus for which the bus is held. If the SCSI bus is disconnected, this bit is ignored.

0 ⇒ If the SCSI bus is being held, defer execution of this command.
1 ⇒ Execute this command to continue the current SCSI operation.

DSD Disconnect - This bit is used to tell the adapter to disconnect from the SCSI bus after a new command is received as a result of either the CMD bit or LSC bit being set.

0 ⇒ This option will not be used.
1 ⇒ The adapter must disconnect after receiving a new command.

BFR Bus Free Phase - Use this bit to cause an immediate Bus Free phase. The adapter will send no message bytes before going to Bus Free phase. This bit can be used to cause an unexpected disconnect.

0 ⇒ This option will not be used.
1 ⇒ The adapter will immediately go to Bus Free phase.

CMD Command Phase - Use this bit to cause the adapter to go to Command phase and receive a new command from the initiator. Use the SMB with this bit if a specified Message byte should be sent before going to Command phase. This bit is only used to recover from unexpected events.

0 ⇒ This option will not be used.
1 ⇒ The adapter will go to Command phase to receive a new command from the initiator.

NSC Normal SCSI command termination - Use this bit to cause the adapter to terminate a normal command. The adapter will send the Status Byte followed by the Message Byte and then go to Bus Free phase.

0 ⇒ This option will not be used.
1 ⇒ The adapter will terminate the command by going to Bus Free phase after sending the Status Byte followed by the Message Byte.

Note
As an example of how to use the bits of the Response Control Flags field, suppose you wish to check the data that the RF3880 receives from an initiator before determining what action to take. You could do the following:
1. On the first *Target Mode Response* command, set bits 1 to 7 to zero—the adapter will perform the requested data transfer and then hold the bus without sending any messages.
2. After determining an action, set bits 1 to 7 as desired (to continue the data transfer or terminate the command) set the CSC bit (bit 0), and issue another *Target Mode Response* command.

DSC Disconnect SCSI command termination - Use this bit when the SCSI command is not yet complete and you wish to send another *Target Mode Response* command. Use of this bit will cause the adapter to send the First Message Byte (usually a Save Data Pointer message) followed by the Message Byte, (usually a Disconnect message) and then go to Bus Free phase.

0 ⇒ This option will not be used.
1 ⇒ The adapter will go to Bus Free after sending the First Message Byte followed by the Message Byte.

LSC Linked SCSI command termination - This bit is used to terminate a linked SCSI command and begin receiving the next command. In this case, the adapter will send the Status Byte followed by the Message Byte and then go to Command Phase.

0 ⇒ This option will not be used.
1 ⇒ The adapter will go to Command phase after sending the Status Byte followed by the Message Byte.

SMB Send Message Byte only - Use this bit to cause the adapter to send the First Message Byte (often a Save Data Pointer message) to the initiator after any data transfer is complete, and not go to Bus Free phase.

0 ⇒ This option will not be used.
1 ⇒ The adapter will send the First Message Byte and will not go to Bus Free phase afterward.

Message Byte

Depending on the setting of the NSC or DSC bits of the Response Control Flags field, this field will contain either a Message byte to send after the Status byte (NSC is one), or the second Message byte to send (DSC is one).

Status or 1st Message Byte

Depending on the setting of the NSC and DSC bits of the Response Control Flags field, this field will contain either a status byte (NSC is one), or the first message byte to send (DSC is one).

Queue Tag Identifier

This byte contains the queue tag identifier assigned to this SCSI command by the initiator.

Queue Tag Message

This byte contains a Simple Queue Tag message (20H) if any queue tag message was received from the initiator.

Target Mode Response Command Status Block

In Chapter 4, a Base Status Block was introduced and explained in general terms. The Status Block for a Target Mode Response command uses some of the same fields, and modifies others. A *Target Mode Response* command status block is differentiated from other status blocks by the TMS bit in the Flags field, and also by the Status Information it contains.

Status Block Format

The format of the *Target Mode Response* command Status Block is as follows:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Cmd Status Flag	Reserved	Error	Flags
08H	Msg Byte RCD	CDB Length	CDB Byte 0	CDB Byte 1
0CH	CBD Byte 2	CDB Byte 3	CDB Byte 4	CDB Byte5

Table 52: First TM Response Command Status Block

A SCSI Command Descriptor Block (CDB) being returned to the Host may have more than six bytes. In that case, a second status block is required to contain the remaining CDB bytes. Its format is as follows:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Reserved		Error	Flags
08H	CBD Byte 6	CBD Byte 7	CDB Byte 8	CDB Byte 9
0CH	CBD Byte 10	CDB Byte 11	Reserved	

Table 53: Second TM Response Command Status Block

Description of Fields

A description of each field in the Status Blocks is provided next.

Command Identifier

This value links a Status Block with a Parameter Block.

Flags

This byte can be polled to ascertain whether, and in what condition, a command completed. It has the following format:

BITS							
7	6	5	4	3	2	1	0
CC	ERR	RTY	0	0	CSB	0	TMS

Table 54: Meaning of Status Block Flags Field

TMS Target Mode Status - This bit indicates that this Status Block is for a *Target Mode Response* command.

0 ⇒ Status was not the result of a *Target Mode Response* command.
1 ⇒ Status is a result of a *Target Mode Response* command.

CSB Continued Status Block - This bit indicates that the current Status Block is the second of two Status Blocks generated because the SCSI Command Descriptor Block (CDB) is more than six bytes.

0 ⇒ Status Block is first (or only) Status Block.
1 ⇒ Status Block is second of two Status Blocks.

RTY Retry - Indicates whether this Pass-through command required retries to complete.

0 ⇒ No retries were required.
1 ⇒ Retries were required to complete.

ERR Error Status - Indicates whether this command completed with an error.

0 ⇒ No error occurred, or if non-zero value exists in the Error field, error was recoverable.
1 ⇒An error occurred.

CC Command Complete - Indicates whether a command has finished.

0 ⇒Command not complete.
1 ⇒ Command complete.

Error

When the adapter detects an error, this byte contains an error code to describe the condition. There are special Target Mode error codes that indicate Target Mode error conditions. These codes are listed along with the other adapter error codes in Appendix A.

Command Status Flags

This byte indicates the status of the current command, and of the SCSI bus. You must use it to check for certain conditions that may have occurred and need specific action. The format is as follows:

BITS							
7	6	5	4	3	2	1	0
Error Sequence Code				CMD	MSG	CHK	DIS

Table 55: Meaning of Command Status Flags Field

DIS Disconnect - The bit indicates whether the adapter has disconnected from the SCSI bus while processing this command.

0 ⇒ The adapter has not disconnected.
1 ⇒ The adapter has disconnected to process this command.

CHK Check Condition - Because of an error condition, the RF3880 may attempt to send a Check Condition status followed by a Command Complete message to the initiator.

0 ⇒ No Check Condition status was sent.
1 ⇒ A Check Condition status and Command Compete message to the initiator was attempted.

MSG Message - The adapter (as Target) may receive a Message from the initiator. This bit indicates whether a Message was sent and can be found in the Message Byte Received field.

0 ⇒ No Message Byte was received.
1 ⇒ The Message Byte Received field contains a valid Message from the initiator.

CMD Command Bytes Received - This bit indicates whether the adapter has placed SCSI Command Bytes into the CDB Received area of the *Target Mode Response* command Status Block(s).

0 ⇒ No SCSI commands were received.
1 ⇒ A SCSI command was received and the CDB Length and CDB Received areas are valid.

Error Sequence Code - The Error Sequence Code tells you, according to the following table, where in the process an unrecoverable error occurred, or a Message Byte was sent from the initiator.

Sequence	Description
0	No message was received.
1	Message received after the RF3880 sent the First Message Byte.
2	Message received after sending the Message Byte.
3	Message received after sending the Queue Tag Message.
4	Message received after a Data Transfer phase.
5	Message received after Selection phase.
6	Message received after sending the Identify message.
7	Message received after a Command phase.
8	Message received in response to SDTR message.
9	Message received in response to WDTR message.

Table 56: Target Mode Error Sequence Codes

Command Descriptor Block Length

This byte indicates the number of command bytes that were received from the initiator, and have been passed on in the CDB fields of the *Target Mode Response* command Status Block. If the CMD bit is clear (in the Command Status Flags field), this field is invalid.

Message Byte Received

At any time, the initiator may assert ATN, and cause the adapter to read in a Message byte in response. The adapter will place the Message Byte in this field and return the proper status. If the MSG bit is clear (in the Command Status Flags field), this field is invalid. Also, the value of the Error Sequence Code must be considered when examining the Message Byte. The Error Sequence Code describes where in the process the Message Byte was sent from the initiator.

SCSI CDB fields

These bytes contain the SCSI 6, 10, or 12 byte command received from the initiator as a result of the CMD bit or the LSC bit in the Target Mode Response command Response Control Flags field being set. If the CMD bit is clear (in the Command Status Flags field), this field is invalid.

7

Board-control Commands

Introduction

There are two types of commands that you will issue to the RF3880 adapter: Pass-through commands, and adapter Board-control commands. This chapter concentrates on Board-control commands.

Board-control commands allow you to communicate set-up information and operational guidelines to the RF3880 adapter, as well as request current information from the board, perform diagnostics, and selectively reset the SCSI bus. This chapter is intended to be used as a look-up reference for these Board-control commands issued to the adapter.

If you need information about the SCSI Pass-through commands available, see a current SCSI, SCSI-2, or SCSI-3 specification and CCS addendum. See Chapter 5 for information about initiating an Initiator Pass-through command, or Chapter 6 for Target Mode Pass-through commands.

Summary

This chapter provides detail about each individual Board-control command:

- The command code
- A description of the command
- Explanation of unique flags and Parameter Block fields
- Examples of command usage where helpful

This chapter is most helpful to someone writing or modifying a driver for the RF3880 product.

Using the Reference

The RF3880 adapter recognizes the type of command it is receiving by the contents of the ID field of its parameter block. If the ID field contains a number between 0 and 15 (indicating a SCSI address) the command is a Pass-through command. If the ID field contains an FFH, the adapter recognizes it as a Board-control command. This chapter explains only the Board-control commands. The ID field will always be set to FFH for these commands. In this chapter it is referred to as the Adapter ID field.

Each of the Board-control commands are completely described in the following pages; each description follows the same format:

- First the Parameter Block format is illustrated for each command. The fields that are valid appear in bold type.
- Next, each field used by the command is explained.
- Last, if there is a special type of Status Block returned by the command, it is displayed and all of its fields explained.

General Status Block

When a Board-control command does not return a special type of Status Block, it is assumed that its Status Block will take the general format. That format is described here for reference.

General Status Block Format

The format of the Status Block returned by most Board-control commands takes this form:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Reserved		Error	Flags
08H	Reserved			
0CH				

Table 57: General Status Block

Description of Fields

Following is a brief description of each of the fields used to return General Status:

Command Identifier

This value is used to match a Status Block to its Parameter Block.

Flags

This field can be polled to ascertain whether, and in what condition, a command completed. It has this format:

BITS							
7	6	5	4	3	2	1	0
CC	ERR	0	0	0	0	0	0

Table 58: Bit-meanings of Flags Field

Note
The Flags byte returned in the Status Block of a Board-control command reserves the bits other than ERR and CC. However, some of the reserved bit fields are used in the Flags byte returned in the Status Block of a Pass-through command. See Chapter 5 and 6 for more information.

ERR Error Status - Identifies when a command completes with an error.

0 ⇒ No error occurred.
1 ⇒ An error occurred.

CC Command Complete - Indicates whether a command has finished.

0 ⇒ Command not complete.
1 ⇒ Command complete.

Error

When an adapter-detected error occurs, this byte contains an error code to describe the condition. See Appendix A.

List of Commands

These are the Board-control commands that you can use to set-up the RF3880 operation in your system.

Command Code	Command Name	Page #
01H	Start Command List	7 - 6
02H	Stop Command List	7 - 8
05H	Identify	7 - 9
06H	Board Statistics	7 - 12
07H	General Option	7 - 15
08H	Unit Options	7 - 18
09H	Diagnostic/Self-test	7 - 24
0AH	Target Mode Enable	7 - 27
0BH	Target Mode Disable	7 - 29
0DH	Abort	7 - 30
10H	SCSI Hard Reset	7 - 31
15H	Board Information	7 - 32
16H	Extended Board Statistics	7 - 41
18H	Extended Unit Options	7 - 45

Table 59: Board-control Commands

Start Command List (01H)

With this command you can instruct the RF3880 adapter to begin operating from a Command List. Since with Command List operation you can attain higher performance from your SCSI peripheral, this is one of the first commands you should issue to the adapter. It must be issued by means of a Single Command Structure.

See Chapter 4, Command Operation for information about using Command Lists.

Command Format

The Parameter Block Format for the Start Command List command looks like this:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Reserved		Addr Mod	Adapter ID = FFH
08H	Command List Memory Address			
0CH	Reserved		Interrupt	
10H	Command = 01H	Reserved		
14H	Reserved			
18H	Reserved			

Table 60: Start Command List Parameter Block

Description of Fields

Each of the fields used by the Start Command List command are explained in the following section:

Command Identifier

The Command Identifier field is used to identify the Status Block associated with a Parameter Block. It must be a unique value.

Adapter ID

This field must always contain the ID of FFH to indicate a Board-control command.

Address Modifier

This is the value that the adapter is to drive onto the VMEbus Address Modifier Lines when selecting the memory address of the Command List.

Command List Memory Address

This field contains the address of the Command List in Memory. It points to the beginning of the structure: Parameter Block IN index.

Interrupt

This field defines the interrupt level for the adapter to use when reporting command completion. You can also denote the resultant interrupt acknowledge ID that the adapter will place on the data bus in response to the host. The Interrupt field of the Start Command List Parameter Block uses these bit-meanings:

BITS															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	INT			Status/ID							

Table 61: Bit-meanings of Interrupt Field

Status/ID Interrupt Status/ID - This byte denotes the value the adapter will provide to identify the interrupt being acknowledged by the Host.

INT Interrupt Level - You may elect to use any one of the following levels for adapter interrupt:

000 — Interrupt disabled
001 — Interrupt level 1
010 — Interrupt level 2
011 — Interrupt level 3
100 — Interrupt level 4
101 — Interrupt level 5
110 — Interrupt level 6
111 — Interrupt level 7

Note
An Error Code of 11H will be returned in the Status Block if you attempt to start an already started Command List.

Command Code

This field indicates the command to be executed. For the Start Command List command it is always equal to 01H.

Stop Command List (02H)

The Stop Command List command is used to gracefully halt Command List operations. Upon receipt of this command, the adapter will accept no further additions to the Command List queue.

After all other existent commands are executed from the Command List, a Status Block for this command is returned.

The Stop Command List command is useful when you are shutting down your system. It processes and returns status of commands current at the time of its issue, therefore you can be sure all data is transferred properly before shutdown.

This command should be issued via a Single Command Structure.

Command Format

The format of the Stop Command List Parameter Block is shown below:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Reserved			Adapter ID = FFH
08H	Reserved			
0CH	Reserved			
10H	Command = 02H	Reserved		
14H	Reserved			
18H	Reserved			

Table 62: Stop Command List Parameter Block

Description of Fields

Each of the fields used by the Stop Command List command are explained in the following section:

Command Identifier

The Command Identifier field is used to identify the Status Block associated with a Parameter Block. It must be a unique value.

Adapter ID

The Stop Command List command is a Board-control command. To identify it as such to the RF3880 adapter, this field must contain the value FFH.

Note
An error code of 11H will be returned in the Status Block for this command if you attempt to stop a Command List that is stopped (or has not been started).

Command Code

The command code to use to indicate the Stop Command List command is 02H.

Identify (05H)

This command returns a special Status Block that contains information about the Firmware installed on the board. It can be used to determine board compatibility and support of options (especially useful if your application depends on certain functions to be in place on every product). The Board Information command, discussed in a later section, returns this information plus Unit Options setup for each target and hardware jumper settings.

The Identify command can be issued with either type of command structure. Its primary use may be during initialization, which may require use of a Single Command Structure (due to limitations on available memory at power-up).

Command Format

The Parameter Block format for the Identify command looks like this:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Reserved			Adapter ID = FFH
08H	Reserved			
0CH	Reserved			
10H	Command = 05H	Reserved		
14H	Reserved			
18H	Reserved			

Table 63: Identify Command Parameter Block

Description of Fields

The Identify command requires only three fields as explained in the following descriptions.

Command Identifier

The Command Identifier field is used to identify the Status Block associated with a Parameter Block. It must be a unique value.

Adapter ID

An FFH indicates that this is a Board-control command.

Command Code

The Command Code for the Identify command is 05H.

Identify Command Status Block

The Status Block returned by the Identify command contains fields that are slightly different from the standard Status Block. The fields of this special Status Block are described below:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	FW Rev.	Eng, Rev.	Error	Flags
08H	Option Flags	Day	Month	Year
0CH	Firmware Number of Even Prom			

Table 64: Identify Command Status Block

Description of Fields

The Status Block returned from an Identify command contains these fields:

Command Identifier

The value in this field will be identical to the Parameter Block Value.

Flags Field

This byte is used to indicate command completion. It has this format:

BITS							
7	6	5	4	3	2	1	0
CC	ERR	0	0	0	0	0	0

Table 65: Bit-meanings of Flags Field

ERR Error Status - Indicates if the command completed with an error.

0 ⇒ No error occurred.
1 ⇒ A command execution error occurred.

CC Command Complete - Indicates a command has finished.

0 ⇒ Command is not complete.
1 ⇒ Command has completed.

Error

When an adapter-detected error occurs, this byte contains an error code to describe the condition. See Appendix A for descriptions of all codes that could occur in this field.

Engineering Revision

This field indicates the revision level used by Ciprico to identify its source code. This field is useful only for Ciprico and should not be used to identify the revision level of the board firmware.

Firmware Revision

This field contains a value that identifies the released level of Firmware for the product.

Option Flags

The Options field of the Identify command Status Block has the following format:

BITS							
7	6	5	4	3	2	1	0
SCSI ID 2	SCSI ID 1	SCSI ID 0	RST	0	1	0	SCSI ID3

Table 66: Option Flags Field - Identify Status Block

RST Reset Jumper - Indicates whether the SCSI bus will be reset at power-up and when adapter is reset through the reset port.

0 ⇒ SCSI bus will not be reset.
1 ⇒ SCSI bus will be reset.

SCSI ID BITS SCSI ID of Adapter - Indicates the current SCSI bus ID of the RF3880 adapter.

0 ⇒ Bit of ID not set.
1 ⇒ Bit of ID set.

SCSI ID	Option Flags Bits			
	0	7	6	5
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

Table 67: Adapter SCSI ID Bits

Day, Month, Year

These three fields, when combined, indicate the date (in hex) that the firmware in the EPROM was compiled for production.

Firmware Number of Even EPROM

There is a set of two EPROMs on the board. This number is the Firmware number of the EVEN EPROM.

Board Statistics (06H)

The RF3880 adapter keeps track of several types of board statistics. This particular command maintains compatibility with earlier RF35XX products; it is used to report three types of statistics. The Extended Board Statistics command, described in a following section, reports additional board statistics.

The statistics are valid only as long as power is kept to the system. This command reports the statistics by recording them in the area of system memory specified in the command Parameter Block. (This statistics structure is returned in addition to the normal Status Block.)

You can clear the internal statistics table at any time by issuing the command with the CLR bit of the Options field set. No data transfer occurs when the CLR bit is set.

Command Format

The Parameter Block format for the Board Statistics command looks like this:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Reserved		Addr. Mod.	Adapter ID = FFH
08H	VME Memory Address			
0CH	Reserved			
10H	Command = 05H	Options	Reserved	
14H	Reserved			
18H	Reserved			

Table 68: Board Statistics Command Parameter Block

Description of Fields

Each of the fields of the Parameter Block are explained below:

Command Identifier

The Command Identifier field is used to identify the Status Block associated with a Parameter Block. It must be a unique value.

Adapter ID

The Board Statistics command is a Board-control command. This field must be set to FFH.

Address Modifier

This is the value to be used by the adapter on the Address Modifier lines when writing statistics to memory. This field is not used when the CLR bit is set.

VME Memory Address

This is the system memory address at which you wish to have the adapter place the board statistics. This field is not used when the CLR bit of the Options byte is set.

Command Code

The value for this field will always be 06H to indicate a Board Statistics command.

Options

This field has the following bit-meanings:

BITS							
7	6	5	4	3	2	1	0
0							CLR

Table 69: Options Field

CLR Clear - This bit allows you to clear the internal statistic values currently stored. (Statistics returned by both the 06H and 16H commands will be cleared.) It is useful if you wish to keep track of statistics during a specific period. The only other time statistic values get cleared is during adapter reset.

0 ⇒ Command is used to return internal statistics to specified system memory address.
1 ⇒ Command is used to clear the internal statistics table. VMEbus Address fields are not used.

The Data Structure returned from the Board Statistics command is illustrated on the next page.

Statistics Data Structure

The Statistics Data Structure returned from a Board Statistics command will have this format:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Total Command Processed			
04H	Total SCSI Commands Issued to ID 0			
08H	Total SCSI Commands Issued to ID 1			
0CH	Total SCSI Commands Issued to ID 2			
10H	Total SCSI Commands Issued to ID 3			
14H	Total SCSI Commands Issued to ID 4			
18H	Total SCSI Commands Issued to ID 5			
1CH	Total SCSI Commands Issued to ID 6			
20H	Total SCSI Commands Issued to ID 7			
24H	Number of Times SCSI Check Conditions Status Received			
28H	Total SCSI Commands Issued to ID 8			
2CH	Total SCSI Commands Issued to ID 9			
30H	Total SCSI Commands Issued to ID 10			
34H	Total SCSI Commands Issued to ID 11			
38H	Total SCSI Commands Issued to ID 12			
3CH	Total SCSI Commands Issued to ID 13			
40H	Total SCSI Commands Issued to ID 14			
44H	Total SCSI Commands Issued to ID 15			
48H	Target Mode Selection Count			
4CH thru 58H	Reserved			

Table 70: Board Statistics Data Structure

Description of Fields

The Board Statistics Data Structure returned from the Board Statistics command contains these fields:

Total Commands Processed

This field reports the number of commands, Pass-through and Board-control, that have been processed by the adapter since power-up or since the CLR option has been used.

Total SCSI Commands issued to an ID

Each of these fields reports the number of commands the adapter has issued to the respective SCSI ID.

Number of SCSI Check Condition Status Received

This field reports the number of times a SCSI device reported a Check Condition status.

Target Mode Selection Count

Number of times the adapter was selected as a target device.

General Options (07H)

The General Options command allows you to select how the RF3880 adapter treats system and SCSI operation. The options you can select are general to the entire SCSI bus. Other options are available to you on a per unit basis with the Unit Options command and Extended Unit Options command.

The options of this command need only be set once during initialization; they will remain in effect until the next General Options command is issued or until the adapter is reset.

Command Format

The Parameter Block for the General Options command looks like this:

Description of Fields

Each field of the Parameter Block is explained below:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Select Flags	Bus Throttle	SCSI Bus ID	Adapter ID = FFH
08H	Reserved			
0CH	Reserved			
10H	Command = 07H	Reserved		
14H	Reserved			
18H	Reserved			

Table 71: General Options Command Parameter Block

Command Identifier

The Command Identifier field is used to identify the Status Block associated with a Parameter Block. It must be a unique value.

Adapter ID

The General Options command is a Board-control command. This value must be FFH.

SCSI Bus ID

Each device that is connected to the SCSI bus must have an ID in order to communicate on the bus; the initial SCSI ID of the RF3880 adapter is set through jumpers. (See Chapter 3 for information on setting the jumpers.)

Note
You can use the Board Information command (7 - 32) to find out what ID the jumpers are set to without physically examining the board.

This field of the General Options command allows you to select a different ID than the one jumpered for the adapter, without changing the jumpers. This can be useful to avoid conflicts caused by duplicated ID’s at power-up.

Valid values are 0 - FH.

Bus Throttle

Note
The adapter will enforce a minimum burst size equal to the transfer width. The adapter will also enforce a maximum burst for Block Mode transfers, in order to ensure adherence to the VME specification—rules 2.12 and 2.75—related to address boundaries.

The bus throttle control allows you to regulate the flow of SCSI data on the VMEbus. The flow of data is regulated by limiting the number of bytes that the adapter is allowed to transfer each time it acquires the bus. The adapter transfers the data in bursts. You may specify the number of bytes that the adapter transfers in a burst by placing the appropriate value in the Bus Throttle field.

The number of bytes in a burst is expressed as a power of 2. The possible values are 0 though 0BH.

The table below translates the Bus Throttle field values into number of bytes per burst.

The default setting for this field is 0BH, 2048 bytes per burst.

Value	Represents	Number of Bytes per Burst
00H	2^0	1
01H	2^1	2
02H	2^2	4
03H	2^3	8
04H	2^4	16
05H	2^5	32
06H	2^6	64
07H	2^7	128
08H	2^8	256
09H	2^9	512
0AH	2^{10}	1024
0BH	2^{11}	2048

Table 72: Bus Throttle Field Values

Select Flags

The Select Flags are used to determine several operating factors for the RF3880 adapter. Use of Block Mode transfers by the adapter is affected by Select Flags.

Other factors pertain to the SCSI bus: parity checking and disconnect/reconnect.

The format of the Select Flags byte is as follows:

BITS							
7	6	5	4	3	2	1	0
0	0	0	0	X	BMT	PAR	DIS

Table 73: Select Flags Byte

DIS Disconnect/Reconnect - With this bit you are able to select whether the RF3880 adapter will allow peripherals to disconnect from the SCSI bus while performing a command.

0 ⇒ Disallow Disconnect/Reconnect.
1 ⇒ Allow Disconnect/Reconnect.

The default value for this bit is one — allow disconnects on all commands.

PAR Check SCSI Bus Parity - This bit allows you to select whether the adapter checks errors in parity that occurred on the SCSI bus. (The default is set by SCSI configuration jumpers. See Chapter 3.)

0 ⇒ Do not check for parity.
1 ⇒ Check for SCSI parity errors.

Note
Use the option that sets the adapter to always perform Block Mode transfers only when you are providing User-defined Address Modifiers that do Block Mode Transfers.

BMT Block Mode Transfers - This bit allows you to force the adapter to perform Block Mode transfers regardless of the Address Modifier used.

0 ⇒ The RF3880 performs Block Mode transfers for only those Address Modifiers listed in the VME specification.
1 ⇒ The RF3880 performs Block Mode transfers regardless of the Address Modifier.

Bit 3 - Please note that Bit 3 has meaning for other Ciprico adapters that are compatible with the RF3880, but this bit is ignored on the RF3880.

Command Code

The command code for this command is 07H.

Unit Options (08H)

The Unit Options command allows you to individually tailor how the RF3880 adapter operates, in either initiator or target mode, with each SCSI device on the bus.

Usually these options need be set only once, at power-up, to create the proper operating environment. Each unit you need to set-up requires a separate Unit Options command.

Command Format

The Parameter Block for the Unit Options command looks like this:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Disconnect Time-out		Unit SCSI ID	Adapter ID = FFH
08H	Select Time-out		Retry Control	Retry Limit
0CH	Reserved	Reserved	Sense Count	Unit Flags
10H	Command = 08H	Reserved		
14H	Sense Byte #	Sense Byte #	Sense Byte #	Sense Byte #
18H	Sense Byte #	Sense Byte #	Sense Byte #	Sense Byte #

Table 74: Unit Options Command Parameter Block

Description of Fields

Each of the fields of the Parameter Block are described below:

Command Identifier

This field is used if you are issuing the Unit Options command through the structure of a Command List. It is used to identify the Status block returned from the command. It should be a unique value.

Adapter ID

This command performs no action on a specific SCSI device. It is a Board-control command and so this field must be FFH.

Unit SCSI ID

This field must contain a number from 0 - FH representing a SCSI device on the bus. Do not use the RF3880 adapter SCSI ID you assign with the General Options command.

Disconnect Time-out

If you have enabled SCSI bus Disconnect/Reconnect with the General Options command (DIS bit of Select Flags is set), this field selects — for the specified unit — the amount of time the adapter will wait for a reconnect. If a reconnect does not occur within the time defined, the adapter will post a time-out error (1FH) in the appropriate command Status Block.

The time appropriate for a disconnect depends on the action and the unit performing the task (seeks, rewinds, etc.). For this reason, the Disconnect Time-out value is tailored per unit with a latitude of 100 milliseconds to almost two hours. You may also choose the option of no time-out for this unit. In that case, the device or SCSI bus must be reset to recover from the failure.

0 ⇒ No time-out occurs; SCSI bus or device must be reset to clear a failure.
Each increment ⇒ 100 milliseconds.

Note
The sequentially recorded nature of tape media is such that positioning for retries requires extra commands. For this reason, do not enable the retry function for SCSI tape devices.

Retry Limit

If retries are enabled in the Retry Control field, (discussed next) this field specifies the maximum number of retries to attempt before reporting an error. Once the retry limit is reached, the error condition reported on the last retry is returned in the Status Block.

You must provide a value in this field or retries will not occur even if enabled (in the Retry Control field).

Retry Control

This field is used to particularize the retry operation for this unit. This field allows you to select the type of errors to retry and how they are reported.

BITS							
7	6	5	4	3	2	1	0
0	0	0	RBE	RCE	RPE	ISB	INT

Table 75: Retry Control

INT Issue Interrupt - When ISB bit is set, allows you to select whether the adapter posts an interrupt for retry Status Block(s) when returned. Valid only when ISB bit is set.

0 ⇒ Do not interrupt for retry Status Block(s).
1 ⇒ Interrupt when retry Status Block(s) returned.

ISB Issue Status Block - Allows you to select that a Status Block be returned for each command retry performed. Enabling this will cause multiple Status Blocks to be returned for a command that requires retries.

0 ⇒ Do not issue a Status Block for each retry.
1 ⇒ Issue a Status Block for each retry.

RPE Retry Parity Errors - Allows you to retry SCSI commands that failed with a parity error.

0 ⇒ Do not retry for parity errors.
1 ⇒ Retry parity errors.

RCE Retry Command Errors - Allows you to select whether the adapter will respond to a device error with a retry of the failed command.

0 ⇒ Do not retry for device errors.
1 ⇒ Retry commands that error from the device.

RBE Retry Bus (SCSI) Errors - Enables retries in the event that a command fails.

0 ⇒ Do not perform Bus Error retries.
1 ⇒ Perform Bus Error retries.

Select Time-out

According to SCSI specifications, a target must respond to initiator selection by asserting the BSY signal in response to SEL, with its ID on the bus. With this field you can select the amount of time that the RF3880 adapter will wait for response before timing out with an error.

The select time-out period is specified in 1 millisecond increments. The default value is 250 (FAH), which selects a period of 250 milliseconds.

The longest time-out period possible is 392 milliseconds, selected with a value of 392 (188H) Any larger value placed in this field will still select 392 milliseconds.

A value of 0 will select the shortest time-out period possible, which is about 1 millisecond.

Unit Flags

The Unit Flags field of the Unit Options Parameter Block is where you can define command operation for the unit. With bit settings you can address the issues of: synchronous data transfer, tagged commands, and use of the ATN signal.

BITS							
7	6	5	4	3	2	1	0
0	TAG	0	0	WID	IAT	SYN	UNQ

Table 76: Unit Flags

Note
If you set the TAG flag (explained below) the adapter will ignore the UNQ bit. Setting the TAG bit implies that you are using a SCSI-2 or SCSI-3 device. The UNQ bit is intended for SCSI-1 devices.

UNQ Untagged Queuing - The SCSI-1 specification allows issue of multiple, simultaneous, commands to a target that has multiple LUN’s, one command per LUN. This bit allows you to select whether the RF3880 will queue commands to a target with multiple LUN’s.

0 ⇒ Send one command at a time to this target.
1 ⇒ Queue commands to this target; it has multiple LUN’s.

SYN Synchronous Negotiation - Either the initiator or the target may introduce negotiation for synchronous operation. How you set this bit determines whether the RF3880 adapter will attempt to negotiate with the device. (Even if you do not set this bit to allow negotiation, if the other device begins negotiation, the adapter will respond.)

0 ⇒ Do not initiate negotiation for synchronous data transfer.
1 ⇒ Negotiate for synchronous data transfer.

IAT Inhibit ATN Signal - With this bit you can select whether the adapter will assert the ATN signal during selection of a target. (Some older SCSI devices do not respond to the ATN signal and may stop executing commands if it is asserted.) Setting this bit will prevent the adapter from negotiating for synchronous transfers, wide transfers, and SCSI-2 features. Parity checking will also be disabled. If you enable any of these features with this bit set, you will get errors.

0 ⇒ ATN will be asserted.
1 ⇒ ATN will not be asserted during target selection.

WID Wide Transfers - Either the initiator or the target may introduce negotiation for wide data transfers. How you set this bit determines whether the RF3880 adapter will attempt to negotiate with the device. (Even if you do not set this bit to allow negotiation, if the other device begins negotiation, the adapter will respond.)

0 ⇒ Do not initiate negotiation for wide data transfer

1 ⇒ Negotiate for wide data transfer.

TAG - The function of this bit varies depending upon whether the adapter is in initiator or target mode.

Initiator Mode
Tagged Queuing - One of the design options offered with the SCSI-2 specification is the capability to issue multiple simultaneous commands to each logical unit. If you have a peripheral that is designed to this specification, you can use this bit to tell the RF3880 adapter to negotiate for tagged queuing. The tag “type” is specified in the Flags-2 field of the Pass-through command Parameter Block. Refer to Chapter 5.

0 ⇒ Tagged queuing operation not negotiated.
1 ⇒ Peripheral supports tagged queuing - negotiate for it.

Target Mode
Selection of Group 2 Commands - During Target Mode operations, the Tag bit controls whether Group 2’s 6- or 10-byte SCSI-2 commands are accepted.

0 ⇒ 6-byte commands accepted; no SCSI-2 devices configured.
1 ⇒ 10-byte commands accepted; SCSI-2 devices configured.

Sense Count

The RF3880 adapter automatically responds to a device Check Condition with a SCSI Request Sense command. (Unless inhibited on a per command basis with the IRS bit of the Flags-1 field which is embedded in the SCSI Pass-through command Parameter Block. See Chapter 5.)

The Request Sense command, issued by the adapter to the device with the Check Condition, can garner Sense information of up to 256 bytes. The number returned depends on the number asked for in the SCSI Request Sense command. The Sense Count field is used to define the number of Sense bytes the adapter will ask for when automatically issuing a Request Sense command.

Valid values for this field are 0 to 32 (20H).

If more than 32 bytes of sequential sense information is requested, the count will be set to 8 and error code A1H will be returned.Otherwise, the value you place in the Sense Count field is used in conjunction with the Selected Sense Bytes fields. There are three ways you can use the Sense Count and Selected Sense Bytes fields to get the result you want:

To receive the default — first eight bytes of Request Sense Data:

- Place a zero value in Sense Count field.
- Place zeroes in all fields of Selected Sense Bytes.

To receive any eight bytes of Request Sense Data from the possible 256:

- Place a zero in the Sense Count field.
- Place a number in each of the eight fields of the Selected Sense Bytes, in the order desired.

To receive any number of the first 32 bytes of Request Sense Data:

- Place the number desired in the Sense Count field.
- Selected Sense Bytes fields are ignored.

If you wish to hand-pick more than eight bytes of Request Sense Data to be returned in a specific order, you must use the Extended Unit Options command; with it you can specify up to sixteen selected Request Sense bytes.

Command Code

The command code for this command is 08H.

Selected Sense Bytes

This part of the Unit Options Parameter Block is a combination of eight fields, each containing the byte number of one of 256 possible SCSI Sense bytes. These fields are only valid when the value of the Sense Count field of this command is zero.

If the Sense Count field contains a 0 and all eight fields that make up the Selected Sense Bytes field are set to 0, the default Sense bytes are returned

The default Sense bytes are bytes 0 to 7. Returned in a Status Block, they would appear as shown below.

Byte Memory Address			
Offset + 0	Offset + 1	Offset + 2	Offset + 3
Command Identifier			
Reserved	SCSI Status	Error	Flags
0 = Class/Code	1 = Segment	2 = SCSI Flags	3 = Info Byte
4 = Info Byte	5 = Info Byte	6 = Info Byte	7 = Ex Length

Table 77: Default Ordering of Sense Bytes

If the Sense Count field is set to 0, but there are non-zero values in the Selected Sense bytes fields, the byte numbers and order you define in the Selected Sense Bytes fields are used to return automatic Request Sense information for this device.

Example Use of Selected Sense Bytes Fields

In this example your device has valuable information, in bytes other than the first eight, that you wish to substitute:

- Byte 12 is an Additional Sense Code.
- Bytes 15 through 17 include additional error information.

Assume bytes 16 and 17 are pointers that you wish to read in as a word. You will arrange them in the group of Sense Bytes to make this easy. For the example, the Sense Count field must be set to zero. The Selected Sense Bytes fields of the Unit Options command would look like this:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Disconnect Time-out		Unit SCSI ID	Adapter ID = FFH
08H	Select Time-out		Retry Control	Retry Limit
0CH	Reserved	Reserved	Sense Count = 0	Unit Flags
10H	Command = 08H	Reserved		
14H	0H (Class/Code)	01H (Segment)	02H (SCSI Flags)	03H (Info Byte)
18H	0CH (Byte 12)	0FH (Byte 15)	11H (Byte 17)	10H (Byte 16)

Table 78: Example Selection of Sense Bytes

Diagnostic/ Self-test (09H)

The Diagnostic/Self-Test command duplicates most of the tests performed at power-up. You can select which tests will be performed. The command returns a special Status Block with testing results.

This command writes test data into board memory.

The adapter will not execute the Diagnostic/Self-Test command until it completes all preceding commands in order to avoid writing over valid data in memory. Also, the adapter will not accept other commands while it is executing Diagnostic/Self-test. When the testing is done, the adapter will service all pending Channel Attentions.

Command Format

The Parameter Block for the Diagnostic/Self-test command has this format:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Reserved			Adapter ID = FFH
08H	Reserved			
0CH	Reserved			
10H	Command = 09H	Test Flags	Reserved	
14H	Reserved			
18H	Reserved			

Table 79: Diagnostic/Self-test Parameter Block

Description of Fields

Each field of the Diagnostic/Self-test Parameter Block is described below:

Command Identifier

The Command Identifier field is used to identify the Status Block associated with a Parameter Block. It must be a unique value.

Adapter ID

This field identifies the type of command being issued. This is a Board-control command. The value in this field must be FFH.

Test Flags

This field is used to determine which of the RF3880 board tests will be performed.

BITS							
7	6	5	4	3	2	1	0
0	QLG	STT	PSI	RAM	186XL	PCS	0

Table 80: Diagnostics Command Test Flags Field

PCS PROM CRC Test - This test performs a CRC of all bytes programmed in the EPROMs, compares it to the CRC stored in the highest EPROM memory location, and reports the result in a Status Block.

0 ⇒ Skip the PROM CRC test.
1 ⇒ Perform the PROM CRC test.

186XL 80186XL Processor Test - This routine verifies that the internal registers of the 80186 can be loaded with a sequence of data patterns, checks the arithmetic and logical data handling for correct results and flags, confirms that the program jump instructions function, plus verifies the 80186 can read and write memory.

0 ⇒ Skip the 80186 Processor test.
1 ⇒ Perform the 80186 Processor test.

RAM 512 Kbyte Data Buffer Test - This test writes various data patterns to the Data Buffer, then reads it back to verify.

0 ⇒ Skip the Data Buffer test.
1 ⇒ Perform the Data Buffer test.

PSI Pipelined System Interface Test - This test writes a variety of data patterns to the registers and counters of the Pipelined System Interface and then reads them back to verify. It also confirms that the address and transfer counters accurately reflect the proper DMA transfer mode. (i.e. adjusted for word or longword values).

0 ⇒ Skip the Pipelined System Interface test.
1 ⇒ Perform the Pipelined System Interface test.

STT SCSI Termination Test - This test checks for a blown SCSI termination power fuse and also verifies a good voltage at TERM PWR pin on the SCSI connector.

0 ⇒ Skip the SCSI Termination test.
1 ⇒ Perform the SCSI Termination test.

QLG QLogic Fast SCSI chip Test - This test writes a progression of data patterns to the registers and counters of the QLogic Fast SCSI chip, then reads each back to verify the pattern.

0 ⇒ Skip the QLogic Fast SCSI chip test.
1 ⇒ Perform the QLogic Fast SCSI chip test.

Command Code

The command code for this operation is 09H.

Status Block Format

The Status Block returned from the Diagnostic/Self-test command looks like this:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Reserved		Error	Flags
08H	Reserved			
0CH	Reserved			

Table 81: Diagnostic Command Status Block

Description of Fields

The Status Block returned from a Diagnostic/Self-test command contains these fields:

Command Identifier

The value in this field will be identical to the Parameter Block Value.

Flags Byte

This byte is used to indicate command completion:

BITS							
7	6	5	4	3	2	1	0
CC	ERR	0	0	0	0	0	0

Table 82: Bit-meanings of Flags Byte in Status Block

ERR Error Status - Indicates whether the command completed with an error.

0 ⇒ No error occurred.
1 ⇒ An error occurred.

CC Command Complete - Indicates if the command has finished.

0 ⇒ Command not complete.
1 ⇒ Command complete.

Error

This field contains error codes specific to the tests performed as well as those error codes generated from system and SCSI bus errors. See Appendix A.

Target Mode Enable (0AH)

With this command you instruct the RF3880 to begin accepting selection from initiators on the SCSI bus. The adapter will still operate as an initiator, but after this command has been issued, it will also operate as a target, until a Target Mode Disable command is sent or a SCSI bus reset occurs.

Command Format

The Parameter Block format for the Target Mode Enable command looks like this:

Note
If you send a Target Mode Response command to the RF3880 before you have enabled it to act in Target Mode with the Target Mode Enable command, the Response command will be terminated with an error code. If you send the RF3880 an Enable command while it is already enabled, the second Enable command will terminate with an error code.

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Reserved	Reserved	Addr Mod	Adapter ID = FFH
08H	Pointer to Target Selection Data Structure			
0CH	Reserved		Selection Interrupt	
10H	Command = 0AH	Test Flags	Reserved	
14H	Reserved			
18H	Reserved			

Table 83: Target Mode Enable Parameter Block

Description of Fields

Each of the fields used by the Target Mode Enable command are explained in the following section:

Command Identifier

The Command Identifier is used to identify the Status Block associated with a Parameter Block. The adapter also writes this value into the Command Identifier area of the Target Selection Data Structure. It must be a unique value.

Adapter ID

This field must contain an FFH to indicate that it is an adapter command.

Address Modifier

This is the value that the adapter is to drive onto the VMEbus Address Modifier lines when selecting the memory address of the Target Selection Data Structure.

Pointer to the Target Selection Data Structure

This field contains the address of the area to be used by the adapter to create a Target Selection Data Structure after being selected by an initiator.

Selection Interrupt

This field is used to define the interrupt level for the adapter to use when information in the Target Selection Data Structure is ready for the Host to read. It also contains the interrupt acknowledge ID that the adapter will place on the data bus in response to the Host. The format is:

BITS															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	INT			Status/ID							

Table 84: Meaning of Selection Interrupt Field

Status/ID Interrupt Status/ID - This byte denotes the value the adapter will provide to identify the interrupt being acknowledged by the Host.

INT Interrupt Level - You may elect to use any one of the following levels for adapter interrupt:

000 — Interrupt disabled
001 — Interrupt level 1
010 — Interrupt level 2
011 — Interrupt level 3
100 — Interrupt level 4
101 — Interrupt level 5
110 — Interrupt level 6
111 — Interrupt level 7

Command Code

This field indicates the command to be executed. For the Target Mode Enable command, it is always equal to 0AH.

Target Mode Disable (0BH)

Use the Target Mode Disable command when you wish the adapter to cease responding to selections from initiators on the SCSI bus. Also with this command, you can choose how the adapter will respond to selections by initiators on the SCSI bus after Target Mode is disabled.

Command Format

The Parameter Block format for the Target Mode Disable command looks like this:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Reserved	Flags-1	Reserved	Adapter ID = FFH
08H	Reserved			
0CH	Reserved			
10H	Command = 0BH	Reserved		
14H	Reserved			
18H	Reserved			

Table 85: Target Mode Disable Parameter Block

Description of Fields

Each of the fields used by the Target Mode Disable command are explained in the following section:

Command Identifier

The Command Identifier is used to identify the Status Block associated with a Parameter Block. It must be a unique value.

Adapter ID

This field must contain an FFH to indicate that it is an adapter command.

Flags-1

BITS							
7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	RBS

Table 86: Meaning of Flags-1 Field

The Flags-1 byte contains specific controls for the operation of the Target Mode Disable command. The format is as follows:

RBS Return Busy on Selection - This bit determines how the board will respond to selection by SCSI initiators after SCSI Target Mode is disabled.

0 ⇒ Respond to future selections with unexpected Bus Free.

1 ⇒ Respond to future selections with Busy status and Command Complete message.

Command Code

This field indicates the command to be executed. For the Target Mode Disable command, it is always equal to 0BH.

Abort (0DH)

This command is used to halt a specific command. It attempts to find a command with the Command Identifier specified, and abort it. If the Abort command is successful, the Abort error code (3BH) will be returned in the Status Block of the command being aborted, and the Abort command will be completed without error. If the command with the specified Command Identifier cannot be found, the Abort command will return a Status Block with the Abort Failed error code (3CH).

Command Format

The Parameter Block format for the Abort command looks like this:

Note
This command must be used to abort pending Target Mode Response commands when an ABORT, ABORT TAG, CLEAR QUEUE, or BUS DEVICE RESET message is received from an initiator.

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Reserved			Adapter ID = FFH
08H	Command Identifier of CMD to be Aborted			
0CH	Reserved			
10H	Command = 0DH	Reserved		
14H	Reserved			
18H	Reserved			

Table 87: Abort Parameter Block

Description of Fields

Each of the fields used by the Abort command are explained in the following section:

Command Identifier

The Command Identifier is used to identify the Status Block associated with a Parameter Block. It must be a unique value.

Adapter ID

This field must contain an FFH to indicate that it is an adapter command.

Command Identifier of CMD to be Aborted

This field contains the Command Identifier of a command that should be aborted. The adapter will search for the first command that has this Command Identifier and will abort it.

Command Code

This field indicates the command to be executed. For the Abort command it is 0DH.

SCSI Hard Reset (10H)

The SCSI Hard Reset command simply asserts the reset (RST) signal on the SCSI bus. It will abort all commands being executed, even those being executed by devices that are currently disconnected. For this reason, it should only be used as a drastic means of recovery.

Command Format

The Parameter Block for the SCSI Hard Reset command has this format:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Reserved			Adapter ID = FFH
08H	Reserved			
0CH	Reserved			
10H	Command = 10H	Reserved	Reserved	
14H	Reserved			
18H	Reserved			

Table 88: SCSI Hard Reset Parameter Block

Description of Fields

Each field of the SCSI Hard Reset Parameter Block is described below:

Command Identifier

The Command Identifier field is used to identify the Status Block associated with a Parameter Block. It must be a unique value.

Adapter ID

This field identifies the type of command being issued. This is a Board-control command. The value in this field must be FFH.

Command Code

The command code for this Board-control command is 10H.

Board Information (15H)

This command returns a special Board Information Data Structure (in addition to the normal status block), that provides information about the current configuration of the RF3880 adapter. The structure, returned at the address you specify in the Board Information command Parameter Block, includes information about the options set with the General Options and Unit Options commands as well as the configuration set in the hardware. It can be used as a snapshot of the board set-up. Additionally, the same information received from the Identify command is included in the structure returned.

This command can be issued with either a Single Command structure or in a Command List.

Command Format

The Parameter Block for the Board Information command has this format:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Reserved		Addr. Mod.	Adapter ID = FFH
08H	VME Memory Address			
0CH	Reserved			
10H	Command = 15H	Options	Reserved	
14H	Reserved			
18H	Reserved			

Table 89: Target Mode Disable Parameter Block

Description of Fields

Each field of the Board Information Parameter Block is described below:

Command Identifier

The Command Identifier field is used to identify the Status Block associated with a Parameter Block. It must be a unique value.

Adapter ID

This is a Board-control command. The value in this field must be FFH.

Address Modifier

This is the value to be used by the adapter on the Address Modifier lines when writing the Board Information Data Structure to memory.

VME Memory Address

This is the system memory address at which you wish to have the adapter place the Board Information Data Structure.

Command Code

The value for this field will always be 15H to indicate a Board Information command.

Options

The Options field is one byte wide with the following format and bit-meanings:

BITS							
7	6	5	4	3	2	1	0
0						FMT	0

Table 90: Board Information Options Field

FMT - This bit determines whether the data returned with the Board Information command will include information about all possible SCSI ID's (16) or will retain compatibility with previous adapters that were limited to 8 ID's on a bus.

0 ⇒Use 8 ID format.
1 ⇒ Use 16 ID format.

The format of the Data Structure returned from the Board Information command is illustrated in the follwoing table.

Board Information Data Structure

The Data Structure returned by the Board Information command has two parts. The first part of the structure consists of a special header primarily containing General Board Information (including hardware configuration).

The second part of the returned structure consists of eight or sixteen blocks of data (dependent on the setting of the FMT bit in the Options field of the Board Information command). These blocks describe the Unit Options that are set for each target attached to the RF3880. The first block corresponds to SCSI ID 0, the second to SCSI ID 1, etc.

Address Offset	Byte Memory Address				
	Offset + 0	Offset + 1	Offset + 2	Offset + 3	
00H	FW Rev.	Eng. Rev.	Bus Throttle	Select Flags	General Board Information Header
04H	Option Flags	Day	Month	Year	
08H	Firmware Number on Even Prom				
0CH	Adapter Base Model #		SCSI Cfig Block	Termination	
10H	Reserved				
14H	Sense Count	Retry Limit	Disconnect Time-out		Per Target Information
18H	Synch Offset	Synch Period	Device Flags		
1CH	Width	Reserved			
20H	Sense Byte #	Sense Byte #	Sense Byte #	Sense Byte #	
24H	Sense Byte #	Sense Byte #	Sense Byte #	Sense Byte #	
28H	Sense Byte #	Sense Byte #	Sense Byte #	Sense Byte #	
2CH	Sense Byte #	Sense Byte #	Sense Byte #	Sense Byte #	
thru 1D0H	Target information repeated up to fifteen more times — depending on setting of Options field (in command).				

Table 91: Board Information Data Structure

Each of the fields of this Data Structure are described above and on the following pages.

Description of Fields

The Data Structure returned from a Board Information command contains the fields shown on the following pages.

Select Flags

This field displays the selections currently valid for the Select Flags field of the General Options command. The format of the Select Flags byte is as follows:

BITS							
7	6	5	4	3	2	1	0
0	0	0	0	0	BMT	PAR	DIS

Table 92: Board Information - Select Flags Field

DIS Disconnect/Reconnect - This bit indicates whether the RF3880 adapter will allow peripherals to disconnect from the SCSI bus while performing a command.

0 ⇒ Disconnect/Reconnect is disallowed.
1 ⇒ Disconnect/Reconnect is allowed.

PAR Check SCSI Bus Parity - This bit indicates whether the adapter reports errors in parity that occurred on the SCSI bus.

0 ⇒ Parity will not be checked.
1 ⇒ SCSI parity will be checked.

BMT Block Mode Transfers - This bit indicates how the adapter is set to perform Block Mode transfers.

0 ⇒ Block Mode transfers will occur only for Address Modifiers listed in the VME specification.
1 ⇒ Block Mode transfers will be performed regardless of Address Modifier used.

Bus Throttle

This field displays the currently selected throttle option in an identical manner as the field used to select throttle in the General Options command. Possible values are shown below:

Value	Represents	Number of Bytes per Burst
00H	2^0	1
01H	2^1	2
02H	2^2	4
03H	2^3	8
04H	2^4	16
05H	2^5	32
06H	2^6	64
07H	2^7	128
08H	2^8	256
09H	2^9	512
0AH	2^{10}	1024
0BH	2^{11}	2048

Table 93: Bus Throttle Field Values

Engineering Revision

This field indicates the revision level used by Ciprico to identify its source code. This field is useful only for Ciprico and should not be used to identify the revision level of the board firmware.

Firmware Revision

This field contains a value that identifies the released level of Firmware for the product.

Day, Month, Year

These three fields, when combined, indicate the date (in hex) that the firmware in the EPROM was compiled.

Option Flags

The Options Flags byte returned here is identical to the byte returned with the Identify command. It is used to indicate whether a particular option is enabled or available in the revision of firmware installed.

The bit-meanings of the Option Flags field are as follows:

BITS							
7	6	5	4	3	2	1	0
SCSI ID 2	SCSI ID 1	SCSI ID 0	RST	0		0	SCSI ID3

Table 94: Option Flags Field - Identify Status Block

RST Reset Jumper

0 ⇒ SCSI bus will not be reset.
1 ⇒ SCSI bus will be reset.

SCSI ID BITS SCSI ID of Adapter - Indicates the current ID of the adapter.

0 ⇒ Bit of ID not set.
1 ⇒ Bit of ID set.

SCSI ID	Option Flags Bits			
	0	7	6	5
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

Table 95: Adapter SCSI ID Bits

Firmware Number of Even EPROM

There is a set of two EPROMs on the board. This number is the Firmware Number of the EVEN EPROM.

Termination

This field returns the status of the SCSI fuse on the RF3880 and the condition of terminator power on the SCSI cable connector. It has this format:

BITS							
7	6	5	4	3	2	1	0
0	0	0	0	0	BSF	BTP	0

Table 96: Bit Meanings of Termination Field

BTP Bad SCSI Termination Power - Indicates whether there is sufficient voltage on the TERM PWR pin of the SCSI connector.

0 ⇒ Proper 4 - 5.25 volts available.
1 ⇒ Less than 4 volts available.

BSF Blown SCSI Fuse - Indicates whether the Termination Power Fuse on the adapter is intact.

0 ⇒ Fuse is good.
1 ⇒ Fuse is blown.

SCSI Cfig Block

This field reports the status of the SCSI Configuration Block of hardware jumpers. You can use it to determine how the jumpers are set without removing the board from your system:

BITS							
7	6	5	4	3	2	1	0
1 - 2	3 - 4	5 - 6	7 - 8	9 - 10	11 - 12	13 - 14	15 - 16

Table 97: SCSI Configuration Jumpers

Each bit corresponds to a hardware jumper.

0 ⇒ Jumper is not present.
1 ⇒Jumper is installed.

The purpose for each jumper is described below:

Pins	Description	Default
1 - 2	Reserved	OUT
3 - 4		OUT
5 - 6	Enable SCSI Parity Checking	OUT = Selected
7 - 8	SCSI bus Reset at Power-up	OUT = Selected
9 - 10	SCSI ID Bit 3	OUT = 0
11 - 12	SCSI ID Bit 2	OUT = 0
13 - 14	SCSI ID Bit 1	OuT = 0
15 - 16	SCSI ID Bit 0	OUT = 0

Table 98: SCSI Cfig Block - Default Jumper Settings

Adapter Base Model

This field of the Board Information command was added for the RF3880. If the board is an RF3880, the value 3880H will be returned.

Per Target - Disconnect Time-out

If Disconnects are enabled for the target with the General Options command, this field represents the amount of time the adapter will wait for a reconnect before taking error recovery action. For more information, refer to the section of this chapter that explains the General Options, Unit Options, and Extended Unit Options commands.

Per Target - Retry Limit

This field specifies the maximum number of retries the RF3880 will attempt before reporting an error. This value is set for each target with the Unit Options or Extended Unit Options command.

Per Target - Sense Count

This field reports the number of Sense bytes that the adapter will return for this target. This number is defined by the Unit Options or Extended Unit Options commands.

Per Target - Device Flags

The Device Flags field is a word wide. It reports the options that you have selected for this target with the Unit Options or Extended Unit Options command. The format is as shown below:

BITS															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	SS	0	RBE	RCE	RPE	ISB	INT	0	TQE	0	0	WID	IAT	SYN	UNQ

Table 99: Bit-meanings of Device Flags Field

UNQ Untagged Queuing - This bit indicates whether the RF3880 will queue commands to a target with multiple LUN’s.

0 ⇒ One command at a time will be sent to this target.
1 ⇒ Commands to this target will be queued; it has multiple LUN’s.

SYN Synchronous Negotiation - This bit indicates whether negotiations for synchronous transfers will be initiated for this target.

0 ⇒ Synchronous transfer negotiation will not be initiated by adapter.
1 ⇒ Adapter will initiate synchronous transfer negotiations.

IAT Inhibit ATN Signal - This bit indicates if the adapter is to assert the ATN signal.

0 ⇒ Adapter will assert ATN for this target.
1 ⇒ Adapter will not assert ATN for this target.

WID Wide Transfers - This bit indicates whether negotiations for wide data transfers will be initiated for this target.

0 ⇒ Wide data transfer negotiations will not be initiated by the adapter.
1 ⇒ Adapter will initiate wide data transfer negotiations.

Note
The setting of this bit indicates only that you have enabled tagged queuing operation for this target using the Unit Options (or Extended Unit Options) command. It does not indicate that the device is actually operating in that mode.

TQE Tagged Queuing Enabled - This bit indicates whether you have enabled tagged queuing operation with this target.

0 ⇒ Tagged Queuing is not enabled.
1 ⇒ Tagged Queuing is enabled.

INT Issue Interrupt - Used with the ISB bit, the INT bit indicates whether the adapter will post an interrupt for retry Status Block(s) returned for this target.

0 ⇒ Will not interrupt for retry Status Block(s).
1 ⇒ Will interrupt when retry Status Block(s) returned.

ISB Issue Status Block - This bit indicates whether a Status Block will be returned for each command retry performed on this target.

0 ⇒ Will not issue a Status block for each retry.
1 ⇒ Will issue a Status Block for each retry.

RPE Retry Parity Errors - This bit indicates whether the adapter will retry SCSI commands that failed with a parity error.

0 ⇒ Will not retry for parity errors.
1 ⇒ Will retry parity errors.

RCE Retry Command Errors - This bit indicates whether the adapter will respond to a device error with a retry of the failed command.

0 ⇒ Will not retry for device errors.
1 ⇒ Will retry commands that error from the device.

RBE Retry (SCSI) Bus Errors - This bit indicates whether the adapter will retry a command that fails.

0 ⇒ Will not retry SCSI Bus Errors.
1 ⇒ Will retry commands that failed due to SCSI Bus Errors.

SS Selected Sense - This bit indicates whether you are using the Selected Sense byte option to return up to 16 bytes of specially ordered and selected Sense data.

0 ⇒ Sense data defaults to first eight bytes returned from the device.
1 ⇒ Selected Sense option in effect; Sense data returned according to selection.

Per Target - Synch Period

The Synchronous Transfer Period is negotiated between initiator and target when the target is able to operate in synchronous mode. The common value is agreed upon via Messages between initiator and target.

The Synchronous Transfer Period is the minimum time allowed between leading edges of each successive REQ pulse and each successive ACK pulse.

The transfer period (in nanoseconds) is calculated by multiplying the negotiated value by 4:

Transfer period(nanoseconds) = negotiated value X 4

Per Target - Synch Offset

This value is negotiated between initiator and target when the target is able to operate in synchronous mode. The common value is agreed upon via Messages between initiator and target. The Synch Offset is the maximum number of REQ pulses allowed to be outstanding before the corresponding ACK pulses are received. This value is usually a function of the buffering available on the device.

Per Target - Width

This value reports the result of negotiations between initiator and target for whether the target will operate in wide mode. A value of 0 for this field indicates the target width is 8 bits. A value of 1 indicates 16 bit mode operation.

Per Target - Selected Sense Bytes

If the Selected Sense option is in effect, these 16 fields indicate which Sense bytes, in which order, will be returned from the device.

If you have enabled 16 Selected Sense Bytes using the Extended Unit Options command you will receive the Selected Sense bytes in two Status Blocks. In that case, the first eight Sense byte Fields displayed in the Board Information Data Structure will be returned with the first Status Block. The last eight will appear in the second Status Block.

Extended Board Statistics (16H)

The RF3880 adapter keeps track of several types of board statistics. With the Extended Board Statistics command you can retrieve the extended set of statistics kept by the adapter. The statistics are valid only as long as power is kept to the system. This command reports the statistics by recording them in the area of system memory specified in the command Parameter Block; this statistics structure is returned in addition to the normal Status Block.

You can clear the internal statistics table at any time by issuing the command with the CLR bit of the Options field set. No data transfer occurs when the CLR bit is set.

Command Format

The Parameter Block format for the Extended Board Statistics command looks like this:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Reserved		Addr. Mod.	Adapter ID = FFH
08H	VME Memory Address			
0CH	Reserved			
10H	Command = 16H	Options	Reserved	
14H	Reserved			
18H	Reserved			

Table 100: Extended Board Statistics Parameter Block

Description of Fields

Each of the fields of the Parameter Block are explained below:

Command Identifier

This field is used to identify the Status Block associated with a Parameter Block. It must be a unique value.

Adapter ID

This field must be set to FFH.

Address Modifier

This is the value to be used by the adapter as an Address Modifier when writing extended statistics to memory. This field is not used when the CLR bit is set.

VME Memory Address

This is the system memory address at which you wish to have the adapter place the Extended Statistics Data Structure. This field is not used when the CLR bit is set.

Options

This field has the following format and bit meanings:

BITS							
7	6	5	4	3	2	1	0
0						FMT	CLR

Table 101: Extended Board Options Field

CLR Clear - This bit allows you to clear the internal statistic values currently stored. (Statistics returned by both the 06H and 16H commands will be cleared.) It is useful if you wish to keep track of statistics during a specific period. The only other time statistic values get cleared is during adapter reset.

0 ⇒ Command is used to return internal statistics to specified system memory address.
1 ⇒ Command is used to clear the internal statistics table.

FMT Format - This bit determines whether the data returned from this command will include information about all possible SCSI ID’s (16) or will retain compatibility with previous adapters that were limited to 8 ID’s on a bus. This bit is ignored if the CLR bit is set.

0 ⇒ Use 8 ID format.
1 ⇒ Use 16 ID format.

Command Code

The value for this field will always be 16H to indicate an Extended Board Statistics command.

Extended Statistics Data Structure

The Extended Statistics Data Structure returned from an Extended Board Statistics command will have this format:

Address Offset	Byte Memory Address				
	Offset + 0	Offset + 1	Offset + 2	Offset + 3	
00H	Total Commands Issued				General Board Information Header
04H	Number of SCSI Bad Status Errors				
08H	Number of SCSI Parity Errors				
0CH	Target Mode Selection Count				
10H	Reserved				
14H	SCSI Target ID 0: Number of Commands				Per Target Information; Targets 0 - 15
18H	Queue Full Count		Maximum Queued		
1CH	Reserved				
.	.				
.	.				
.	.				
C8H	SCSI Target ID 15: Number of Commands				
CCH	Queue Full Count		Maximum Queued		
D0H	Reserved				

Table 102: Extended Board Statistics Data Structure

Description of Fields

The Extended Statistics Data Structure returned from the Extended Board Statistics command contains these fields:

Total Commands Issued

This field reports the number of commands, SCSI Pass-through and Board-control, that have been processed by the adapter since power-up, reset, or the CLR bit option was used.

Number of SCSI Bad Status Conditions

This field reports the number of times a status other than Good was reported by all SCSI devices.

Number of SCSI Parity Errors

This field reports the total number of SCSI Parity Errors that have occurred since power-up, reset, or the CLR bit option was used.

Target Mode Selection Count

This field reports the number of times the RF3880 was selected as a target device.

Per SCSI Target — Number of Commands

This field is repeated for each of the 16 possible SCSI Target ID’s. It reports the number of commands received by that target since power-up, reset, or the CLR bit option was used. (The Number of Commands field for the SCSI ID that corresponds to the adapter ID will be zero, as will any ID in the table that is not presently used by a device.)

Per SCSI Target — Queue Full Count

This field reports the number of times a target reported a Queue Full status. It may be useful in evaluating the performance of tagged queuing operation with a SCSI-2 or SCSI-3 target.

Per SCSI Target — Maximum Queued

This field reports the maximum number of commands that were queued to a target and may also be useful in evaluating the performance of SCSI-2 (and SCSI-3) tagged queuing operation.

Extended Unit Options (18H)

This command, in either initiator or target mode, is designed as an enhancement to the Unit Options command. The primary difference between the commands is the opportunity to select 16 bytes of Sense Data to be returned per unit, instead of the eight offered in the Unit Options command.

Another difference is that the Extended Unit Options for the SCSI device are built into a Data Structure whose address is simply passed in the command.

Command Format

The format of the Extended Unit Options command is as follows:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Reserved		Addr. Mod.	Adapter ID = FFH
08H	VME Memory Address			
0CH	Reserved			
10H	Command = 18H	Reserved		
14H	Reserved			
18H	Reserved			

Table 103: Extended Unit Options Parameter Block

Description of Fields

Each of the fields of the Parameter Block are described below:

Command Identifier

The Command Identifier field is used to identify the Status Block associated with a Parameter Block. It must be a unique value.

Adapter ID

This is a Board-control command. The value in this field must be FFH.

Address Modifier

This is the value to be used by the adapter on the Address Modifier lines when reading the Extended Unit Options Data Structure from memory.

VME Memory Address

This is the system memory address at which you have built the Extended Unit Options Data Structure.

Command Code

The value for this field will always be 18H to indicate a Extended Unit Options command.

The format of the Data Structure to be used for the Extended Unit Options command is illustrated on the following pages.

Extended Unit Options Data Structure

The Data Structure that you will use to pass the Extended Unit Option parameters to the RF3880 adapter is similar to the format of the Unit Options Parameter Block.

The Data Structure must contain these fields:

Description of Fields

Each of the fields of the Data Structure are described below:

Address Offset	Byte Memory Address				
	Offset + 0	Offset + 1	Offset + 2	Offset + 3	
00H	Disconnect Time-out		Unit SCSI ID	0	
04H	Select Time-out		Retry Control	Retry Limit	
08H	Reserved	Reserved	Sense Count	Unit Flags	
0CH	Sense Byte #	Sense Byte #	Sense Byte #	Sense Byte #	Status Block #1
10H	Sense Byte #	Sense Byte #	Sense Byte #	Sense Byte #	
14H	Sense Byte #	Sense Byte #	Sense Byte #	Sense Byte #	Status Block #2
18H	Sense Byte #	Sense Byte #	Sense Byte #	Sense Byte #	
1CH	0				

Table 104: Extended Unit Options Data Structure

Unit SCSI ID

This field must contain a number from 0 - 15 representing a SCSI Unit on the bus. Do not use the RF3880 adapter SCSI ID you assign with the General Options command. (If you do you will receive the 02H (Bad Unit) error code.)

Disconnect Time-out

If you have enabled SCSI bus Disconnect with the General Options command, this field selects — for the specified unit — the amount of time the adapter will wait for a reconnect. If a reconnect does not occur within the time defined, the adapter will post a time-out error (1FH) in the appropriate command Status Block.

The time appropriate for a disconnect depends on the action and the unit performing the task (seeks, rewinds, etc.). For this reason, the Disconnect Time-out value is tailored per unit with a latitude of 100 milliseconds to almost two hours. You may also choose the option of no time-out for this unit. In that case, the device or SCSI bus must be reset to recover from the failure.

0 ⇒ No time-out occurs; SCSI bus or device must be reset to clear a failure.
Each increment ⇒ 100 milliseconds.

Note
The sequentially recorded nature of tape media is such that positioning for retries requires extra commands. For this reason, do not enable the retry function for SCSI tape devices.

Retry Limit

If retries are enabled in the Retry Control field, (discussed next) this field specifies the maximum number of retries to attempt before reporting an error. Once the retry limit is reached, the error condition reported with the last retry is returned in the Status Block.

You must provide a value in this field or retries will not occur even if enabled (in the Retry Control field).

Retry Control

This field is used to particularize the retry operation for this unit. This field allows you to select the type of errors to retry and how they are reported.

BITS							
7	6	5	4	3	2	1	0
0	0	0	RBE	RCE	RPE	ISB	INT

Table 105: Bit-meanings of Retry Control Field

INT Issue Interrupt - When ISB bit is set, allows you to select whether the adapter posts an interrupt for retry Status Block(s) when returned. Valid only when ISB is set.

0 ⇒ Do not interrupt for retry Status Block(s).
1 ⇒ Interrupt when retry Status Block(s) returned.

ISB Issue Status Block - Allows you to select that a Status Block be returned for each command retry performed. Enabling this will cause multiple Status Blocks to be returned for a command that requires retries.

0 ⇒ Do not issue a Status Block for each retry.
1 ⇒ Issue a Status Block for each retry.

RPE Retry Parity Errors - Allows retries of SCSI commands that failed with a parity error.

0 ⇒ Do not retry for parity errors.
1 ⇒ Retry parity errors.

RCE Retry Command Errors - Allows you to select whether the adapter will respond to a device error with a retry of the failed command.

0 ⇒ Do not retry for device errors.
1 ⇒ Retry commands returning bad SCSI status.

RBE Retry Bus (SCSI) Errors - Enables retries in the event that a command fails.

0 ⇒ Do not perform Bus Error retries.
1 ⇒ Perform Bus Error retries.

Select Time-out

According to SCSI specifications, a target must respond to initiator selection by asserting the BSY signal in response to SEL, with its ID on the bus. With this field you can select the amount of time that the RF3880 adapter will wait for response before timing out with an error.

The Select Time-out period is specified in increments of 1 millisecond. The default value is 250 (FAH), which selects a period of 250 milliseconds.

The longest time-out period possible is 392 milliseconds, selected with a value of 392 (188H). Any larger value placed in this field will still select 392 milliseconds.

A value of 0 will select the shortest time-out period possible, which is about 1 millisecond.

Unit Flags

The Unit Flags field of the Extended Unit Options Parameter Block is where you can define command operation for the unit. With bit settings you can address the issues of: synchronous data transfer, tagged commands, and use of ATN signal.

BITS							
7	6	5	4	3	2	1	0
0	TAG	0	0	WID	IAT	SYN	UNQ

Table 106: Unit Flags of Extended Unit Options

Note
If you set the TAG flag (explained on next page) the adapter will ignore the UNQ bit. Setting the TAG bit implies that you are using a SCSI-2 or SCSI-3 device. The UNQ bit is intended for SCSI-1 devices.

UNQ Untagged Queuing - The SCSI-1 specification allows issue of multiple, simultaneous, commands to a target that has multiple LUN’s, one command per LUN. This bit allows you to select whether the RF3880 will queue commands to a target with multiple LUN’s.

0 ⇒ Send one command at a time to this target.
1 ⇒ Queue commands to this target; it has multiple LUN’s.

SYN Synchronous Negotiation - Either the initiator or the target may introduce negotiation for synchronous operation. How you set this bit determines whether the RF3880 adapter will attempt to negotiate with the device. Even if you do not set this bit to allow negotiation, if the other device begins negotiation, the adapter will respond.

0 ⇒ Do not initiate negotiation for synchronous data transfer.
1 ⇒ Negotiate for synchronous data transfer.

IAT Inhibit ATN Signal - With this bit you can select whether the adapter will assert the ATN signal during selection of a target. (Some older SCSI devices do not respond to the ATN signal and may stop executing if it is asserted.) Setting this bit will prevent the adapter from negotiating for synchronous transfers, wide transfers, and SCSI-2 and SCSI-3 features. If you enable any of these features with this bit set, you will get errors.

0 ⇒ ATN will be asserted.
1 ⇒ ATN will not be asserted during target selection.

WID Wide Transfers - Either the initiator or the target may introduce negotiation for wide data transfers. How you set this bit determines whether the RF3880 adapter will attempt to negotiate with the device. (Even if you do not set this bit to allow negotiation, if the other device begins negotiation, the adapter will respond.)

0 ⇒ Do not initiate negotiation for wide data transfer
1 ⇒ Negotiate for wide data transfer.

TAG - The function of this bit varies depending on whether the adapter is in Initiator or Target Mode.

Initiator Mode
Tagged Commands - In initiator Mode, one of the design options offered with the SCSI-2 and SCSI-3 specification is the capability to issue multiple simultaneous commands to each logical unit. If you have a peripheral that is designed to this specification, you can use this bit to tell the RF3880 adapter to negotiate for queued tagged commands. The tag “type” is specified in the Flags-2 field of the Pass-through command Parameter Block. Refer to Chapter 5.

0 ⇒ Tagged commands not negotiated.
1 ⇒ Peripheral supports tagged commands - negotiate for it.

Target Mode
Selection of Group 2 Commands - During Target Mode operations, the Tag bit controls whether Group 2 6- or 10-byte SCSI-2 commands are accepted.

0 ⇒ 6-byte commands accepted; no SCSI—2 devices configured.
1 ⇒ 10-byte commands accepted; SCSI—2 devices configured.

Sense Count

The RF3880 adapter automatically responds to a device Check Condition with a SCSI Request Sense command. (Unless inhibited on a per command basis with the IRS bit of the Flags-1 field which is embedded in the SCSI Pass-through command Parameter Block. See Chapter 5.)

The Request Sense command, issued by the adapter to the device with the Check Condition, can garner Sense information of up to 256 bytes. The number returned depends on the number asked for in the SCSI Request Sense command that is issued by the adapter. The Sense Count field is used to define the number of Sense bytes the adapter will ask for when automatically issuing a Request Sense command.

Valid values for this field are 0 to 32 (20H).

If more than 32 bytes of sequential sense information is requested, the count will be set to 8 and error code A1H will be returned.

Otherwise, the value you place in the Sense Count field is used in conjunction with the Selected Sense Bytes Fields. There are three ways you can use the Sense Count and Selected Sense Bytes fields to get the result you want:

To receive the default —first eight bytes of Request Sense Data

- Place a zero value in Sense Count field.
- Place zeroes in all fields of Selected Sense Bytes.

To receive any sixteen bytes of Request Sense Data from the possible 256:

- Place a zero in the Sense Count field.
- Place the Sense Byte number in each of the sixteen fields of the Selected Sense Bytes, in the order desired.

To receive any number of the first 32 bytes of Request Sense Data (any number over eight will generate multiple Status Blocks for each Request Sense command the adapter issues.):

- Place the number desired in the Sense Count field.
- Selected Sense Bytes fields are ignored.

Selected Sense Bytes

This part of the Extended Unit Options Parameter Block is a combination of sixteen fields, each containing the byte number of one of 256 possible SCSI Sense bytes. These fields are only valid when the value of the Sense Count field of the Data Structure is zero.

If the Sense Count field contains a 0 and all sixteen fields that make up the Selected Sense Bytes section of the Data Structure are set to 0, the default of eight Sense bytes is returned.

The eight default Sense bytes are displayed as follows:

Byte Memory Address			
Offset + 0	Offset + 1	Offset + 2	Offset + 3
Command Identifier			
Reserved	SCSI Status	Error	Flags
0 = Class/Code	1 = Segment	2 = SCSI Flags	3 = Info Byte
4 = Info Byte	5 = Info Byte	6 = Info Byte	7 = Ex Length

Table 107: Default Ordering of Sense Bytes

If the Sense Count field is set to 0, but there are non-zero values in the Selected Sense bytes fields, the byte numbers and order you define in the Selected Sense Bytes fields are used to return automatic Request Sense information for this device.

If you define more than eight bytes to be returned as Sense Data, you will receive two SCSI Pass-through command Status Blocks.

See Chapter 5 for more information about the format of the Status Blocks that will be returned.

8

Details of Usage

Introduction

The previous chapters informed you about the details you will need to install the board, set up command structures, and issue commands to the board and the peripherals attached to it. This chapter will provide information you can use to make decisions about how the RF3880 adapter can be used in your system. It explains in detail how and why to use some of the features of the adapter.

Summary

This chapter explains how to use the following operations for the RF3880 adapter:

- Byte and Word swapping of command structures.
- Tagged Queuing.
- Target Mode.

This chapter is useful to use as you are making decisions about your system and set-up.

Byte and Word Swapping

The RF3880 adapter is flexible enough to allow you to request Byte Swapping, Word Swapping, or both, during transfers of command structures.

This section will first give you some examples to explain why the swapping feature is offered to you, then provide a perspective on implementation with the RF3880 for both Command Structures used by the RF3880 (which the adapter can swap) and data (which needs special software if swapping is necessary).

Need for Swapping

The memory architecture of your system depends on the type of processor you are using. There are several major manufacturers that each use different arrangements of bytes, words, and double-words in memory. The two most common formats are: Intel and Motorola. Their differences are illustrated in the example below:

Example of Memory Differences

If you were to define the following variables in a program:

```
char byte[] = “test”;
int word[2] = {0x0102, 0x0304};
double dword = 0xAABBCCDD;
```

They would appear this way in memory (beginning at location n) for each respective format:

variable	Byte # of memory	Intel Order				Motorola Order			
byte	n	t	e	s	t	t	e	s	t
word	n + 4	02	01	04	03	01	02	03	04
dword	n + 8	DD	CC	BB	AA	AA	BB	CC	DD

Table 108: Variations in Intel and Motorola Memory

Essentially the two processors store a string of bytes in the same order. However, the bytes that make up word and double-word information are stored in an order opposite to each other:

The RF3880 adapter must correctly access the command structures for either type of processor.

Processor Type	Byte of word or double-word	Order in memory
Motorola	LSB*	stored in highest memory address
	MSB*	stored in lowest memory address
Intel	LSB	stored in lowest memory address
	MSB	stored in highest memory address

Table 109: Explanation of Memory Storage Differences

Throughout this manual, the Parameter and Status Blocks that make up the command structures are represented double-word wide, in Motorola ordering.

If you have an Intel processor, you will need to translate these displays to your format. The following example will help explain this.

Example Pass-through Command

The information in the table below would be required to complete a parameter block that issues a Pass-through command. The example shown below uses:

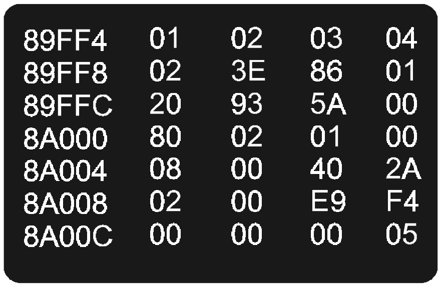
- Logical block size is 128 bytes per block.
- Extended Write SCSI command.
- Command issued to LUN 1 of Target 2.
- Transferring data from VME memory address 5A9320H.
- Transferring data to Logical block 8F4E9H on the device.
- Parameter Block written to memory address 89FF4H.

Example parameters for a Pass-through command:

Name of Parameter Block Field	Value Used for Example	Comments
Command Identifier	04030201H	
Target ID	02H	
Address Modifier	3EH	
Flags-1	86H	
Flags-2	01H	
VME Memory Address	005A9320H	
Transfer Count	00010280H	
SCSI Command Byte 0	2AH	Extended Write Command
SCSI Command Byte 1	40H	Logical Unit 1
SCSI Command Byte 2	00H	Logical Block Address (MSB)
SCSI Command Byte 3	08H	Logical Block Address
SCSI Command Byte 4	F4H	Logical Block Address
SCSI Command Byte 5	E9H	Logical Block Address (LSB)
SCSI Command Byte 6	00	Reserved
SCSI Command Byte 7	02H	Transfer Count (MSB)
SCSI Command Byte 8	05H	Transfer Count (LSB)
SCSI Command Byte 9	00	Control Byte
SCSI Command Byte 10	00	Not Used
SCSI Command Byte 11	00	Not Used

Table 110: Pass-through command Parameters

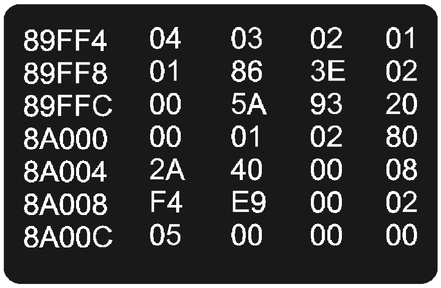
If written to VME memory in the order given above, these command parameters would be written differently (as shown below) for both types of processors:



```
89FF4  01  02  03  04
89FF8  02  3E  86  01
89FFC  20  93  5A  00
8A000  80  02  01  00
8A004  08  00  40  2A
8A008  02  00  E9  F4
8A00C  00  00  00  05
```

00000555.001

Figure 19 Intel Ordering



```
89FF4  04  03  02  01
89FF8  01  86  3E  02
89FFC  00  5A  93  20
8A000  00  01  02  80
8A004  2A  40  00  08
8A008  F4  E9  00  02
8A00C  05  00  00  00
```

00000556.001

Figure 20 Motorola Ordering

It becomes apparent when examining the memory displays, that the Command Structures of the RF3880 though divided into a variety of byte, word, and double-word fields, must be considered as a grouping of double-words.

In a Motorola environment, the Parameter Block would remain as graphically represented throughout this manual, and as shown below:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier - 04030201H			
04H	Flags-2 = 01H	Flags-1 = 86H	Addr. Mod = 3EH	Target ID = 02H
08H	VME Memory Address - 005A9320H			
0CH	Transfer Count = 00010280H			
10H	0 = 2AH	1 = 40H	2 = 0	3 = 08H
14H	4 = F4H	5 = E9H	6 = 0	7 = 02H
18H	8 = 05H	9 = 0	10 = 0	11 = 0

Table 111: Motorola Ordered Parameter Block

However, if you have an Intel processor, you must reinterpret the representations in this manual to an order like this:

Note
Offset + x values are in opposite order for Intel Processors

Address Offset	Byte Memory Address			
	Offset + 3	Offset + 2	Offset + 1	Offset + 0
00H	Command Identifier - 04030201H			
04H	Flags-2 = 01H	Flags-1 = 86H	Addr. Mod = 3EH	Target ID = 02H
08H	VME Memory Address - 005A9320H			
0CH	Transfer Count = 00010280H			
10H	0 = 2AH	1 = 40H	2 = 0	3 = 08H
14H	4 = F4H	5 = E9H	6 = 0	7 = 02H
18H	8 = 05H	9 = 0	10 = 0	11 = 0

Table 112: Intel Ordered Parameter Block

In order for this transition to properly occur, more than just your interpretation of the graphics in the manual must change. The RF3880 must be told to also reinterpret the structures it uses.

RF3880 Swapping Options

Byte and Word Swapping methodology for the RF3880 adapter is configured with the Control field of the Address Buffer Port. This field is in the first word written to the Address Buffer Port. (For bit-specific information on the Address Buffer Port see Chapter 2, Hardware Essentials.)

Swapping is a function of the firmware on the RF3880 adapter and is enabled and performed with minimal impact on performance.

There are two bits in the Control field that allow you to separately control byte swapping and word swapping. If you set neither of the swapping bits in the Control field, the Command Structures will be ordered for a standard Motorola processor.

Command Structure Swapping

As explained, the need to byte-swap and word-swap the fields of the RF3880 Command Structures is due to the differences inherent in various processor architectures. Since the RF3880 defaults to using Motorola ordering for its structures, if your system uses a Motorola processor, no byte and word swapping is necessary. However, if your system uses an Intel processor, the Command Structures of the RF3880 must be adjusted.

This could be done entirely in software — the Host processor could re-order the Command Structures. However this extra burden for the system is not necessary because, by simply enabling the Byte and Word Swap controls, the RF3880 adapter will automatically re-order the Command Structures.

Data Swapping

Since the RF3880 does not handle data swapping you may need to provide software that will do the swapping when you are sharing data between two different types of processors.

Sharing Tapes

For example, if you are writing tapes with an Intel based machine and only Intel based machines will be reading them, no data swapping is necessary. However, if a Motorola machine was to read the tape, special software to reorder the data structures may be needed.

Dual Initiators

Another case in which data swapping may be needed would be when dual RF3880's are used in separate machines but share the same SCSI bus. If both systems use the same disk media, and each is based on a different processor, one of the systems may need special software to swap the data.

Tagged Queuing

The SCSI-2 and SCSI-3 specification provide a protocol for queuing commands at the device level. The device is responsible for managing the order of the commands, basing the order on message information from the initiator; the tag message used by the initiator instructs the device where to place the command in its queue. The initiator also uniquely numbers each command to identify it.

The RF3880 fully supports the tagged queuing protocol. This section describes the RF3880 implementation of tagged queuing and how to use it.

How to Use Tagged Queuing

The RF3880 implementation of this SCSI-2 and SCSI-3 option is very simple to use. These are the steps:

1. Enable Tagged Queuing

In order to use the SCSI-2 and SCSI-3 option of Tagged Queuing, you must be sure that it is enabled on both the adapter and SCSI device.

Note
If your SCSI-2 or SCSI-3 device is operating with tagged queuing, the Queue Full Count and Maximum Queued fields of the Extended Board Statistics command (16H) will contain values. See Chapter 7.

A. Enabling on the RF3880

For the RF3880, tagged queuing is enabled for use with a device by issuing a Unit Options or Extended Unit Options command with the TAG bit of the Unit Flags field set before you issue any commands you want to tag. The RF3880 will only send tagged messages to a device if this flag has been set. This could be done during initialization.

B. Enabling on a SCSI Device

You may also need to enable tagged queuing on your SCSI device with a Page 0AH Mode Select. See your vendor-supplied documentation for more information. The RF3880 will not send an error to the Host if your device rejects the queue tag message.

2. Choose a tag message.

There are three tag messages specified by the SCSI-2 and SCSI-3 protocol. You need only set the appropriate TAG OPTS bits in the Flags-2 field of the Pass-through Parameter Block for the command you are issuing to the SCSI device.

BITS							
7	6	5	4	3	2	1	0
0	0	0	0	JSM	IAD	TAG OPTS	

Table 113: Meaning of Flags-2 Field

These are the messages defined by the SCSI-2 and SCSI-3 specification, their meaning, and the TAG OPTS bit assignments for the Flags-2 field of the RF3880 command:

TAG OPT Value	Meaning
00	SIMPLE QUEUE TAG MESSAGE: The command may be processed by the SCSI-2 or SCSI-3 device out of sequence for efficiency. Example: device may sort commands by logical block address.
01	ORDERED QUEUE TAG MESSAGE: The command must be executed by the SCSI-2 or SCSI-3 device in the order sent by the adapter (in order with respect to other commands sent with the Ordered Queue Tag Message.
10	HEAD OF QUEUE TAG MESSAGE: The command is executed by the SCSI-2 or SCSI-3 device before all other commands previously sent, including other Head of Queue Tag commands but excluding the command currently in process.

Table 114: Meaning of TAG OPTS Bits

As you determine the types of tag messages you wish to use, you should be aware that there are rules of execution that your SCSI-2 or SCSI-3 device must follow for each tag message or combination of tag message commands. For example, determination of priority in the case of mixed tag messages, which executes first — an ORDERED QUEUE TAG or a SIMPLE QUEUE TAG? This situation and others are detailed in your SCSI-2 or SCSI-3 specification.

3. The RF3880 sends the tag message.

The RF3880 decodes these two bits, assigns a tag number and then manages the protocol for sending the proper message for the command.

How It Works

This is how the RF3880 processes commands that use tag messages:

1. The tag message is sent after the identify message and before the SCSI command.
2. The adapter sends the commands in the exact order received from the host.
3. The queue tag number that is sent with the tag message is assigned by the adapter in ascending order beginning with 1, maximum of 255. When 255 is reached, the RF3880 starts over again with 1.

Notes on Usage

The following provides additional detail about how tagged queuing works for the RF3880.

Queue Tag Numbers

- The RF3880 will never have duplicate queue tag numbers assigned at the same time to any single device.
- The maximum number out to a device at one time will be 255.
- If all 255 tag numbers have been assigned to commands, the RF3880 simply waits for previous commands to complete before sending any more tagged commands to that device.

Queue Full Status

- If a device reports Queue Full status before it has received all possible 255 commands, the RF3880 sets an internal threshold flag, (MAXIMUM_COMMANDS) equal to the number of commands out at the time of the Queue Full status.
- The command that generated the Queue Full status will once more be sent out to the device when the number of commands out to the device drops below the threshold set in MAXIMUM_COMMANDS. (One or more disconnected commands complete). This cuts down on unnecessary SCSI bus activity.
- Once below the MAXIMUM_COMMANDS threshold, additional commands will continue to be sent to the SCSI device until another Queue Full status is reported, at which time the MAXIMUM_COMMANDS threshold is set again.

Message Reject

- On the initial command to a SCSI device for which tagged queuing has been enabled, the adapter verifies that the device accepted the queue tag message.
- If the device rejects the queue tag message, the RF3880 does not send further queue tag messages in order to avoid unnecessary SCSI bus activity.
- In the case of a message reject of a tag message, no error is reported to the host.

Tagged Queuing Statistics

The Extended Board Statistics command (16H) returns two fields, per target, that may be useful in evaluating the performance of tagged queuing with a SCSI-2 or SCSI-3 device. The fields are: Queue Full Count and Maximum Queued. See Chapter 7 for details.

Target Mode Usage and SCSI Protocol Handling

Writing a driver for Target Mode requires a strong familiarity with the SCSI protocol. This section explains what you can expect from the adapter under certain SCSI conditions. This section addresses the following issues:

- Selection Queue
- SCSI Bus Reset Handling
- Target Mode Disable Command Handling
- Abort Command Handling
- Synchronous Data Transfer Requests
- Wide Data Transfer Requests
- Unexpected Messages Received from the Initiator
- Automatic Adapter Responses
- Disconnects
- Message Handling

Selection Queue

In order to buffer SCSI bus activities from Host events, selection information (from initiators on the SCSI bus), destined for the Target Selection Data Structure, is queued on the adapter. Information from about sixty initiator selections can be stored in the Selection Queue. This means that at the time of a SCSI bus reset, or a Target Mode Disable or Abort command, the queue may have initiator selection information that has not yet been processed by the Host.

SCSI Bus Reset Handling

To avoid a timing window with the host, Target Mode will be terminated upon receipt of a SCSI bus reset. The host may then re-enable the Target Mode when it is ready to continue receiving commands from other initiators. During the time between a bus reset and being re-enabled, the adapter will respond to selection attempts with Busy status.

Upon reset of the SCSI bus, all targets must terminate all commands in process. The RF3880 will do the following:

- Clear the Selection Queue.
- Set the current Target Selection Data Structure Error field to 27H, and clear the TME bit in the Flags field.
- Return Status Blocks for all outstanding Target Mode Response commands with an Error Code of 27H.

At this time the Host should do the following:

- Discard all outstanding initiator selections.
- Process and discard Status Blocks from Target Mode Response commands.
- Re-enable Target Mode when all processes are cleared.

If a SCSI bus reset occurs after a Target Mode Disable command has been sent to the RF3880 (at the time the Host is waiting for the TME field of the Target Selection Data Structure to reflect disabled Target Mode), the following can be expected:

- The Selection Queue will have an entry destined for the Target Selection Data Structure that has the TME bit clear. There may be other entries in the Selection Queue completing before this information gets written to the Target Selection Data Structure. The Host should process these until receipt of the Target Selection Data Structure associated with the Target Mode Disable command.
- Status Blocks will begin to be returned with 27H set in the Error field.
- Target Mode Response commands issued at this time will not be processed by the RF3880—Status Blocks with an Error field value of 27H will be returned.

Target Mode Disable Command Handling

When the Target Mode Disable command is issued to the RF3880, the adapter will do the following:

- Send a Status Block from the Target Mode Disable command with good status.
- Place after the last entry in the Selection Queue, an entry with the TME bit clear.
- Stop accepting new selections from initiators.

The Host should do the following:

- Continue to process entries in the Target Selection
- Data Structure until receipt of the entry with the TME bit clear.
- Wait for all Target Mode Response command Status Blocks to complete.

Abort Command Handling

Event messages occur on the SCSI bus that require the Host to discard a command. The Abort command is used to do this. When the Abort command is issued to the RF3880, the adapter will attempt to locate a command with the indicated Command Identifier. If the command is found, and hasn’t been sent to the initiator/target, it is aborted.

Synchronous Data Transfer Requests

When Synchronous Data Transfers are enabled, (SYN bit set to 1, of the Unit Flags byte of either the Unit Options, 08H, or Extended Unit Options, 18H, commands), the adapter will send a Synchronous Data Transfer Request (SDTR) message whenever it determines that the data transfer agreement is in an indeterminate state. The adapter will initiate the exchange of SDTR messages immediately before starting the data transfer phase. The adapter will also send a SDTR message in response to a SDTR message received from the initiator.

Wide Data Transfer Requests

When Wide Data Transfers are enabled, (WID bit set to 1, of the Unit Flags byte of either the Unit Options, 08H, or Extended Unit Options, 18H, commands), the adapter will send a Wide Data Transfer Request (WDTR) message whenever it determines that the data transfer agreement is in an indeterminate state. The adapter will initiate the exchange of WDTR messages immediately before starting the data transfer phase. The adapter will also send a WDTR message in response to a WDTR message received from the initiator.

Unexpected Messages Received from the Initiator

If an abort, bus device reset, or related message is received (during Target Mode command processing), it will be passed to the host as an unexpected event. The adapter will not disconnect from the SCSI bus until instructed to do so by the host. After receiving the message, the host issues the necessary adapter commands (Abort command) followed by a Target Mode Response command containing a Bus Free instruction to release the SCSI bus.

Automatic Adapter Responses

The adapter will automatically respond to several events without input from the host. These events are:

- Adapter-detected parity errors
- A received Initiator-detected Error message
- A received Message Parity Error message

The adapter will retry the operation in progress whenever any of these events occur. The adapter will continue to retry the operation until the configured retry limit is exceeded (adapter Unit Options command).

Once the retry limit is reached, the adapter will attempt to send a status of Check Condition followed by a message of Command Complete. If a status byte has been sent, a Restore Pointers message is sent. This means that when the retry limit for a command is reached, the command can end in either of two ways:

- A status byte followed by a message
- A message followed by a status byte followed by a message

If the status/message cannot be sent without error, the adapter will go to the Bus Free phase (an unexpected disconnect). The Host will be notified of this occurrence and should prepare sense data for the initiator (Contingent Allegiance Condition).

The adapter will also automatically respond to the initiator’s data transfer negotiation requests:

- Synchronous Data Transfer Request (SDTR) message.
- Wide Data Transfer Request (WDTR) message.

Parity error conditions

If a parity error is detected on incoming message bytes, the adapter will receive all of the remaining message bytes (until ATN is negated) and then request the message bytes again. During the retry, the state of the ATN line will not be monitored; it is required that the initiator respond correctly to this retry sequence.

If a parity error is detected on incoming command or data bytes, the adapter will stop the transfer and send a Restore Pointers message to the initiator. The transfer will then be restarted. Target Mode Response commands must include a Save Data Pointer message before disconnecting to ensure that this retry technique will work.

Message parity errors

If a Message Parity Error message is received from the initiator in response to a message sent to it, the adapter will return to the Message In phase and resend the entire message.

If a Message Parity Error message is received at any other time, the adapter will go to the Bus Free phase and report this illegal message to the host.

Initiator-detected Error messages

If an Initiator-detected Error message is received from the initiator right after a data transfer, the transfer will be halted and the Restore Pointers message will be sent to the initiator. The transfer will then be restarted. It is required that the host always send a Save Data Pointer message before disconnecting to ensure that this retry technique will work.

If an Initiator-detected Error message is received from the initiator right after the Status phase, the Restore Pointers message will be sent to the initiator. The status byte will then be resent.

If an Initiator-detected Error message is received from the initiator right after any other phase, the message will be treated as an unexpected message and will be sent to the host for action.

Target Routines

The adapter does not support target routines. The adapter will not detect the receipt of an Identify message for a target routine. The host should send a Check Condition status if the initiator attempts to select a target routine.

Disconnect Privilege

Initiators that access the adapter as a target should allow disconnect and limit the use of the control field's link bit in their SCSI commands. By not allowing disconnection or using the link bit, the SCSI bus is monopolized by this single command. Any other device wishing to use the bus during this time must wait for it to be released.

Note
It is also possible to send a command that requires the target device to access another SCSI device before completion of the original command. If the original command did not grant the disconnect privilege, the target will hang waiting for the SCSI bus.

Message Handling

On the following pages Message Handling protocol for the adapter is described. There are three areas in which Message Handling is important:

- Message Bytes received from the Initiator
- Status or Message Sequences Sent to the Initiator (by the Host)
- Status or Message Sequences Sent to the Initiator (by the Adapter)

Each of these topics is covered in the following pages.

Message Bytes Received from the Initiator

The following paragraphs describe the messages that may be received from an initiator. The expected action is stated and the adapter's strategy to handle the message is described.

Abort - Clear the current I/O process plus any queued I/O process for this target/LUN. Go to the Bus Free phase after receipt of this message. This message is sent to the host while holding the SCSI bus. It is the responsibility of the host to instruct the adapter to go to the Bus Free phase after all necessary commands are aborted. If only a target exists, go to the Bus Free phase. In this case, do not send this message to the host.

Abort Tag - Clear the current I/O process for this I_T_x nexus. Go to the Bus Free phase after receipt of this message. This message is sent to the host while holding the SCSI bus. It is the responsibility of the host to instruct the adapter to go to the Bus Free phase after all necessary commands are aborted.

Bus Device Reset - Clear all I/O processes on the target side of the system (this is a hard reset condition). Go to the Bus Free phase after receipt of this message. This message is sent to the host while holding the SCSI bus. It is the responsibility of the host to instruct the adapter to go to the Bus Free phase after all necessary commands are aborted.

Clear Queue - Treat this command as a series of Abort messages that are received from each initiator. This message is sent to the host while holding the SCSI bus. It is the responsibility of the host to instruct the adapter to go to the Bus Free phase after all necessary commands are aborted.

Identify - A single identify message must be sent to the adapter each time it is selected as a target device. The adapter will send a Message Reject message to the initiator if it sends an Identify message at any other time.

Initiator Detected Error - If this message occurs after a Data phase or a Status phase, the target retries the current activity. The Restore Pointers message will be sent first. This message is handled by the adapter up to the retry limit specified in the adapters configuration data. If this message occurs after any other phase, it is sent to the host using the Message Byte Received byte.

Message Parity Error - The last message byte sent to the initiator had a parity error. Return to the Message In phase and resend the entire message that had the detected parity error. If the target is not sending message bytes to the initiator and this message comes in, the target will go to the Bus Free phase (a catastrophic error). In this case, the message will be sent to the host so it may create the correct sense data. This message is handled by the adapter up to the retry limit specified in the adapter’s configuration data.

Message Reject - The last message byte sent was inappropriate or not implemented. If the target is not sending message bytes when this message is received, it will reject this message. This message is sent to the host while holding the SCSI bus. It is the responsibility of the host to instruct the adapter what to do next.

No Operation - The initiator does not currently have a valid message to send. Forget about it and continue. This can happen if the target does not respond to an attention condition until a later phase. The adapter will ignore this message and resume whatever action was in process when the ATN was received.

Head of Queue Tag (2 bytes) - Place this I/O process first in the LUN’s command queue. This message is sent to the host in the queue tag message field of the selection command. If less than the required 2 bytes are received, the adapter will send the initiator a message reject.

Ordered Queue Tag (2 bytes) - Place this I/O process into the LUN’s command queue to be executed in the order received. This message is sent to the host in the queue tag message field of the selection command. If less than the required two bytes are received, the adapter will send the initiator a message reject.

Simple Queue Tag (2 bytes) - Place this I/O process into the LUN’s command queue. This message is sent to the host in the queue tag message field of the selection command. If less than the required two bytes are received, the adapter will send the initiator a message reject.

Synchronous Data Transfer Request (5 bytes) - Compare the requested transfer period and the REQ /ACK offset with the limits established on the adapter. If the limits are OK, send back these same values in another SDTR message. If the limits are not OK, modify them and send another SDTR message. If the adapter does not allow synchronous data transfer, send a Message Reject message. The adapter will handle the exchange of these messages. The limits for the adapter are part of its configuration data. If less than the required five bytes are received, the adapter will send the initiator a Message Reject.

Wide Data Transfer Request (4 bytes) - Compare the requested bus width with the width established on the adapter. If they are the same, send back the same value in another WDTR message. If they are different, adopt the smaller of the two and send this value in another WDTR message. The adapter will handle the exchange of these messages. The width for the adapter is part of its configuration data. If less than the required four bytes are received, the adapter will send the initiator a Message Reject.

Unexpected Messages - These are defined as any messages not listed and described above. The adapter will send a Message Reject message.

Reserved Messages - The adapter sends a Message Reject message.

Vendor-Unique Messages - The adapter sends a Message Reject message.

Status or Message Sequences Sent to the Initiator (by the Host)

The following descriptions include all of the status/message sequences that may be sent to the initiator by the host. Any sequence not listed in the following paragraphs is reserved by Ciprico and must not be sent to the adapter (unpredictable results will occur).

Status Byte and Command Complete Message Byte - A status byte followed by a command complete message is to be sent to the initiator. The target will go to the Bus Free phase as soon as this message is sent.

Save Data Pointer Message Byte and Disconnect Message Byte - Directs the initiator to copy the active data pointer to the saved data pointer for the I/O process. The connection is going to be broken, but a later reconnect will be required to complete the process. The target is going to the Bus Free phase as soon as this message is sent.

Status Byte and Linked Command Complete Message Byte - A status byte followed by a Linked Command Complete message is to be sent to the initiator. The target will request the next command from the initiator as soon as this message is sent.

Status byte and Linked Command Complete (with Flag) - A status byte followed by a Linked Command Complete (with Flag) message is to be sent to the initiator. The target will request the next command from the initiator as soon as this message is sent.

Message Reject - The last message sent to the host from the initiator is either inappropriate or has not been implemented. The message will be ignored. This message must be sent in a response command with no data transfer specified.

Simple Queue Tag (2 bytes) - This message revives a I_T_L_Q nexus for a disconnected I/O process. It will be sent by the adapter immediately following the Identify message. The host will send the Simple Queue Tag on all Target Mode Response commands that are caused by a Target Mode selection that used Queued Tagging. The adapter will send this specified message whenever it must reselect the initiator during the processing of the Response command.

Restore Pointers - Directs the initiator to copy the most recently saved command, data, and status pointers for the I/O process to the corresponding active pointers. This message must be sent in a Target Mode Response command with no data transfer specified.

Status or Message Sequences Sent to the Initiator (by the Adapter)

The following descriptions include all of the status/message sequences that may be sent to the initiator by the adapter.

Busy Status Byte and Command Complete Message Byte - This sequence is sent if the adapter is selected by an initiator and the host has not enabled the Target Mode. This should inform the initiator to try again later.

Disconnect Message Byte - The disconnect message is sent just before going to the Bus Free phase after a command descriptor block is received. The adapter always disconnects (if allowed by the Identify message) to free the SCSI bus while the command is being processed by the host.

Identify - This message will be sent to the initiator as part of a reselection sequence. The LUN field of the response command will be used to supply the LUNTRN field of this message.

Message Reject - This message is sent whenever the adapter receives a message that is not allowed.

Restore Pointers - This message will be used to retry either detected parity errors or Initiator Detected Errors. This message is sent during data transfers, status phase, and command phase.

Synchronous Data Transfer Request (5 bytes) - This message will be sent to the initiator in response to a Synchronous Data Transfer Request received from the initiator, or to initiate a Synchronous Data Transfer negotiation.

Wide Data Transfer Request (4 bytes) - This message will be sent to the initiator in response to a Wide Data Transfer Request received from the initiator, or to initiate a Wide Data Transfer negotiation.

A

Error Codes

Introduction

When the RF3880 adapter encounters a command or operation error, it will return a value in the Error byte of the Status Block. This value can be used to determine the nature of the problem the adapter encountered. A list of these error codes is included in this appendix along with the error codes returned by Self-test.

Summary

This appendix provides information on the following:

- Error Codes returned in the Error field of the Status Block.
- Error Codes returned in the Status Port as a result of Self-test.

This appendix is helpful for someone writing a driver for use with the RF3880.

Status Block Error Codes

The Error Codes on the following pages are the codes that you will see returned in the Error field of the Status Block.

The Status Block has this basic format:

Address Offset	Byte Memory Address			
	Offset + 0	Offset + 1	Offset + 2	Offset + 3
00H	Command Identifier			
04H	Reserved	SCSI Status	Error	Flags
08H	SCSI Sense Bytes			
0CH				

Table 115: Base Status Block

The Error Codes are meant to give you an indication of the conditions surrounding an adapter-detected error. There are two types of errors that can occur: system operation errors and SCSI bus errors. Both types of codes are fully described in order to assist you in determining the reason for a failure.

Hexadecimal values that do not appear in the current list of Error Codes are reserved by Ciprico for future use.

Codes Reported in Status Block and Status Port

Codes in the following list that are asterisked (*XXH) are considered catastrophic errors that will also be reported through the Status Port.

Code	Name	Description
01H	Invalid Command	This code will be reported for the following conditions: **In a Board-control Parameter Block**: code used in the Command field is not a valid Board-control command. Occurs when Target/Adapter ID field is equal to FFH (indicating a Board-control command). **In a Pass-through Parameter Block**: An invalid tag type was specified in the Flags-2 field, or an invalid count or address was specified. **In a Message Pass-through Parameter Block**: A non-supported message was specified for a SCSI operation.
02H	Bad Unit Number	This code will be reported for the following conditions: **Target/Adapter ID field**: The value specified in the Target/Adapter ID field of a Parameter Block was not valid (valid values are 00H-0FH and FFH). **In a Pass-through Parameter Block**: The value specified in the Target ID field is the same as the SCSI ID assigned to the RF3880. This is not a valid SCSI ID to issue commands to. **In a Pass-through Parameter Block**: the value used in the LUN field of the SCSI command contained in the Parameter Block is invalid. (Valid values are 00H-07H.) **In General Options Parameter Block**: the value specified in the SCSI Bus ID field (used to designate an RF3880 SCSI ID other than that set with jumpers) is invalid. (Valid values are 00H-0FH.) **In Unit Options or Extended Unit Options Parameter Block**: Value specified in the Unit SCSI ID field was invalid. (Valid values are any number between and including 00H and 0FH, not being used by the RF3880.)
0FH	Bad Command List Size	This code will be reported when the **Start Command List** command terminates for the following reasons: - Number of Parameter Blocks field contains invalid value. (Valid values are 2-2034.) - Number of Status Blocks field contains invalid value. (Valid values are 2 -4096.)
*10H	Bad Command List Index	This code is returned under these conditions: - Parameter Block In (PBIN) index exceeded the number of Parameter Blocks that can be in the list. - Status Block Out (SBOUT) index exceeded the number of Status Blocks that can be in the list. - A VME bus error occurred while accessing the Status Block list.
*11H	State wrong for Start/Stop Command List command	This code is returned under these conditions: **Start Command List** command issued when a command list is already active. **Stop Command List** command issued when no command list is presently active. **Command List Channel Attention** issued when no Command List is active.
*14H	General Firmware Timeout	The adapter times each of its transfers; it will report this error if a VMEbus transfer takes too long to complete. On the VMEbus, this error can occur for transfers of parameters as well as transfers of data.
*15H	VMEbus Error Occurred	A VMEbus error was detected while the adapter was transferring either parameters or data.
1DH	SCSI General Timeout	For Pass-through operations, this error will be reported if a SCSI operation takes too long to complete.
1EH	SCSI Select Timeout	For Pass-through operations, this code is reported when no SCSI device responds to the selection sequence within the timeout period. The timeout period is specified in the Unit Options command or Extended Unit Options command.
1FH	SCSI Disconnect Timeout	For Pass-through commands, this code is reported when a disconnected SCSI device does not reselect the adapter within the timeout period. The timeout period is specified in the Unit Options command or Extended Unit Options command.
20H	SCSI Parity Error	For Pass-through operations, this code is reported for any SCSI information transfer (message, command, data or status) that results in a parity error

Table 116: Error Codes

Code	Name	Description
21H	Unexpected Disconnect	If a SCSI device unexpectedly goes to the bus free state, this code will be reported.
23H	Unit Returned Bad SCSI Status	If the device returns a bad status (status other than 0) in response to a Pass-through command, this code will be reported. The SCSI Status field of the Status Block will contain the Status Byte returned by the device (Busy, Reservation Conflict, Check Condition, etc.). If the SCSI status is Check Condition, and the automatic request sense feature of the adapter is not inhibited, the Selected Sense Bytes fields of the Status Block will contain the Status Bytes returned by the device. These will further describe the Check Condition. If the automatic Request Sense feature is inhibited (with the IRS bit of the Flags-1 byte), the Selected Sense Bytes fields of the Status Block will contain zeroes.
24H	Unexpected SCSI Phase Entered	This code is reported if the SCSI device being accessed enters a phase that is unexpected for the command sequence issued.
25H	Data Transfer Truncated	If an automatic Request Sense command results in fewer bytes transferred than required to satisfy the Selected Sense Bytes fields of the Status Block, this code will be reported. The Selected Sense Bytes fields of the Status Block that have been truncated will contain value FFH to aid the host in determining which Selected Sense Bytes are valid.
27H	SCSI Bus Reset Asserted or Detected	This code can be reported for either of the following reasons: 1.) Certain conditions will arise that will force the adapter to assert RST on the SCSI bus in order to get to bus free. The SCSI command that was executing at the time of the RESET and any commands that were disconnected at the time of the RESET will report this error. A SCSI bus RESET is a last-resort measure that the adapter uses only if all prior error recovery attempts to get the SCSI bus to a free state have failed. 2.) The adapter detected a SCSI bus reset. The SCSI command currently executing and any commands that were disconnected at the time of the reset will report this code.
2BH	SCSI Invalid Data Transfer Condition	For SCSI data transfer commands, if the SCSI device transfers an odd number of bytes, then 1.) changes phases and/or disconnects, and then 2.) subsequently returns to the data transfer phase to continue the transfer, the RF3880 will abort the command and report this error.
31H	Message Clear Queue or Device Reset Occurred	After executing a **Message Pass-through** operation of Bus Device Reset or Clear Queue, the adapter clears its internal queue of commands that were disconnected when the message was sent. This code will be reported in the Status Block of each disconnected command that was terminated due to the message-only operation.
32H	State wrong for Enable/Disable Target Mode Enable Command	This code is returned under these conditions: - Target Mode Enable command issued when Target Mode is already enabled. - Target Mode Disable command issued when Target Mode is not enabled.
33H	Selection Attempted Without Arbitration	An initiator tried to select the RF3880 without an arbitration phase.
34H	Target Mode Message Received	A message byte was received while in Target Mode. This message byte is in the Message Byte Received field of the Selection Data Structure, or the Status Block. The Error Sequence field provides additional information. The adapter will hold the SCSI bus waiting for a response from the Host.
35H	Target Mode Illegal Message Received	The adapter received and rejected an illegal message byte from the initiator. This message byte received is in the Message Byte Received field of the Selection Data Structure, or the Status Block. The Error Sequence field provides additional information. The adapter will disconnect and terminate the SCSI I/O process. This error is used to inform the Host of the action completed. The Host should prepare Sense Data.

Table 116: Error Codes

Code	Name	Description
36H	Target Mode Message Out Limit Exceeded	The adapter received and rejected unsupported or illegal message bytes from the initiator until the Message Out Phase limit was exceeded.
37H	Target Mode Message Protocol Error	The initiator violated message protocol by sending an Identify and only one more byte, or by holding ATN after a message that is required to be the last in a sequence.
39H	SCSI Chip Target Mode Command Timeout	A Target Mode command to the FAS256 did not complete within the allowed time.
3AH	SCSI Chip Target Mode Command Rejected	A Target Mode command to the FAS256 was rejected because of a gross error or an illegal trap.
3BH	Command Abort	This command was terminated by the RF3880 Board-control Abort command.
3CH	Abort Command Failure	The Abort command could not abort the specified command.
8EH	Bad Task	This error code indicates the occurrence of an Internal Firmware error detected by the adapter.
*96H	Internal Firmware Error	This error code indicates that a Firmware error was detected during the execution of a command.
A1H	Invalid Unit Options Sense Count Value	A value greater than 32 was placed in the Sense Count field of a Unit Options or Extended Unit Options command. Valid values for this field are 0 to 32.

Table 116: Error Codes

Status Port Error Codes

In addition to the Error Codes that are reported in the Status Block of the command, there is a group of codes that are used to report Self-test anomalies and also Catastrophic Error Conditions. These are reported in the Status Port.

Catastrophic Error Codes

Catastrophic Error Codes are returned in the Status Port because conditions are such that it is unlikely that a Status Block could be read. The Status Port format at these times is as shown below:

BIT															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Code								0			ERR	0	0	RDY	ENT

Table 117: Status Port: General

The Catastrophic Error Code field can be seen above.

The codes that can be returned in this field of the Status Port are described below:

Code	Name	Description
10H	Bad PBIN or SBOUT Value	The command List PBIN or SBOUT value exceeded the value set for PBNUM or SBNUM.
11H	State wrong for Star/Stop Command List command	This code is returned under these conditions: **Start Command List** command issued when a command list is already active. **Stop Command List** command issued when no command list is presently active. **Command List Channel Attention** issued when no Command List is active.
12H	Target Mode Error	A **Target Mode Selection Area Read Channel Attention** issued when no selection area was written by adapter.
14H	Software VMEbus Timeout	The adapter times each of its VMEbus transfers; it will report this error if a VMEbus transfer takes too long to complete. This can occur for transfers of parameters as well as transfers of data.
15H	VMEbus Error Occurred	A VMEbus error was detected while the adapter was transferring either parameters or data.
96H	Internal Firmware Error	This error code indicates that a Firmware error was detected during the execution of a command.

Table 118: Status Port Catastrophic Error Codes

Self-test Error Codes

Self-test occurs each time the RF3880 is powered on or reset. If one of the Self-tests fail, an Error Code is reported in the Status Port. The format of the Status Port is as follows:

BIT															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0								Self-test Code (STC)						RDY	STC

Table 119: Status Port: Reset

The Self-test Code (STC) fields are highlighted in the illustration above.

If one of the self-tests does not properly complete, its test number will be left in the Self-test Code fields. The table on the next page lists all the Self-test codes and describes the test performed.

Note that bit 1 of the Status Port is not set until the board is in a Ready condition. Therefore, during Self-test it will remain zero.

Diagnostic/ Self-test Command

The Diagnostic/Self-test Board-control command allows you to selectively perform the board Self-tests. In the event that one of the Self-tests does not complete properly, the Error Codes shown on the next page will be used to report the area of difficulty.

It is important to note, however, that the codes are reported in the Error field of the Status Block returned from the Diagnostics/Self-test command. In that case, these codes are returned in the Status Block Error field in addition to the normal Status Block Error Codes.

Code	Description
00H or FDH	The onboard processor is not working; it cannot access firmware EPROM; cannot access the Status Port.
04H	Static RAM error. All words of static RAM are written with FFFFH, then zero. This test is not exhaustive.
0CH	Firmware CRC error. The calculated CRC for odd and even EPROMs does not match the value stored in EPROM.
80H	SCSI PSI Address/Counter registers not masked to WORD boundaries.
84H	SCSI PSI Address/Counter registers no masked to DWORD boundaries.
88H	Value Miscompare errors of SCSI PSI R/W registers in Manual Load mode.
8CH	Value Miscompare errors of SCSI PSI R/W registers in Auto Load mode.
90H	VME PSI Address/Counter registers not masked to WORD boundaries.
94H	VME PSI Address/Counter registers not masked to DWORD boundaries.
98H	Value Miscompare errors of VME PSI R/W registers in Manual Load mode.
9CH	Value Miscompare errors of VME PSI R/W registers in Auto Load mode.
A8H	SCSI termination power fuse is blown, but termination power is still present at the SCSI connector.
ACH	SCSI termination power fuse is blown and termination power is not available at the SCSI connector.
B0H	Value Miscompare error of QLogic Fast SCSI chips R/W registers.
B4H	QLogic Fast SCSI chip did not generate a reset interrupt.
B8H	QLogic Fast SCSI chip registers not zeroed after reset.
BCH	SCSI bus hung, waiting for reset from QLogic Fast SCSI chip. Could be cabling.
D0H	Video RAM buffer error.
E5H	The onboard processor failed the accumulator test.
E8H	The onboard processor failed the store instruction test.
E9H	The onboard processor failed the shift instruction test.
ECH	The onboard processor failed the compare instruction test.
EDH	The onboard processor failed the jump instruction test.

Table 120: Self-test Status Port Error Codes

B

Cables and Connectors

Introduction

The SCSI specification allows devices to operate with either a single-ended or a differential SCSI interface. The difference between interfaces is the manner in which SCSI signals are driven on the cable.

The differential interface uses two lines for each signal (+SIGNAL and -SIGNAL). A signal is true when +SIGNAL is more positive than -SIGNAL. This interface provides better noise immunity than the single-ended interface and so allows a longer cable length to be used.

The single-ended interface uses one line for each SCSI signal. All devices on a single SCSI bus must operate with the same interface.

The RF3880 is available with a 16-bit differential interface, using a 68-pin high density shielded “P” connector.

Summary

This appendix provides the following information:

- Pin-outs of the SCSI connector used for the RF3880 adapter.
- Cable Length allowed for the differential SCSI interface.
- Pinouts of the P1 and P2 VMEbus backplane connectors.

This appendix is most useful for those integrating devices into a system.

SCSI Connector

The RF3880 supports the differential SCSI interface by use of the 68-pin “P” cable connector. Its connector pin-outs are as shown in the table below.

Maximum Cable Length

Maximum cable length for differential operation is 25 meters (or 82 feet). This length includes internal cabling and cable stubs.

Signal Name	Pin Number	Cable Conductor Number		Pin Number	Signal Name
+DB (12)	1	1	2	35	-DB (12)
+DB (13)	2	3	4	36	-DB (13)
+DB (14)	3	5	6	37	-DB (14)
+DB (15)	4	7	8	38	-DB (15)
+DB (P1)	5	9	10	39	-DB (P1)
Ground	6	11	12	40	Ground
+DB (0)	7	13	14	41	-DB (0)
+DB (1)	8	15	16	42	-DB (1)
+DB (2)	9	17	18	43	-DB (2)
+DB (3)	10	19	20	44	-DB (3)
+DB (4)	11	21	22	45	-DB (4)
+DB (5)	12	23	24	46	-DB (5)
+DB (6)	13	25	26	47	-DB (6)
+DB (7)	14	27	28	48	-DB (7)
+DB (P)	15	29	30	49	-DB (P)
Diffsens	16	31	32	50	Ground
Termpwr	17	33	34	51	Termpwr
Termpwr	18	35	36	52	Termpwr
Reserved	19	37	38	53	Reserved
+ATN	20	39	40	54	-ATN
Ground	21	41	42	55	Ground
+BSY	22	43	44	56	-BSY
+ACK	23	45	46	57	-ACK
+RST	24	47	48	58	-RST
+MSG	25	49	50	59	-MSG
+SEL	26	51	52	60	-SEL
+C/D	27	53	54	61	-C/D
+REQ	28	55	56	62	-REQ
+I/O	29	57	58	63	-I/O
Ground	30	59	60	64	Ground
+DB (8)	31	61	62	65	-DB (8)
+DB (9)	32	63	64	66	-DB (9)
+DB (10)	33	65	66	67	-DB (10)
+DB (11)	34	67	68	68	-DB (11)

Table 121: Cable Connector Pin-outs for the RF3886

VMEbus Connector Pinouts

The VMEbus backplane uses two connectors to carry the signals used by the VME protocol. The pin assignments for each are as follows:

Pin Number	Row A	Row B	Row C
1	+D00	-BBSY	+D08
2	+D01	Not Used	+D09
3	+D02	Not Used	+D10
4	+D03	-BG0IN	+D11
5	+D04	-BG0OUT	+D12
6	+D05	-BG1IN	+D13
7	+D06	-BG1OUT	+D14
8	+D07	-BG2IN	+D15
9	GND	-BG2OUT	GND
10	Not Used	-BG3IN	-SYSFAIL
11	GND	-BG3OUT	-BERR
12	-DS1	-BR0	-SYSRESET
13	-DS0	-BR1	-LWORD
14	-WRITE	-BR2	+AM5
15	GND	-BR3	+A23
16	-DTACK	+AM0	+A22
17	GND	+AM1	+A21
18	-AS	+AM2	+A20
19	GND	+AM3	+A19
20	-IACK	GND	+A18
21	-IACKIN	Not Used	+A17
22	-IACKOUT	Not Used	+A16
23	+AM4	GND	+A15
24	+A07	-IRQ7	+A14
25	+A06	-IRQ6	+A13
26	+A05	-IRQ5	+A12
27	+A04	-IRQ4	+A11
28	+A03	-IRQ3	+A10
29	+A02	-IRQ2	+A09
30	+A01	-IRQ1	+A08
31	-12V	Not Used	+12V
32	+5V	+5V	+5V

Table 122: P1 Connector

Pin Number	Row B
1	+5V
2	GND
3	-RETRY
4	+A24
5	+A25
6	+A26
7	+A27
8	+A28
9	+A29
10	+A30
11	+A31
12	GND
13	+5V
14	+D16
15	+D17
16	+D18
17	+D19
18	+D20
19	+D21
20	+D22
21	+D23
22	GND
23	+D24
24	+D25
25	+D26
26	+D27
27	+D28
28	+D29
29	+D30
30	+D31
31	GND
32	+5V

Table 123: P2 Connector

C

Specifications

Introduction

The RF3880 is designed to meet certain physical, electrical, and environmental criteria. Its specifications are listed on the following page.

Summary

This appendix lists the pertinent specifications for the RF3880 SCSI host bus adapter.

This appendix should be useful to anyone who is responsible for evaluating/ selecting system hardware.

Specifications

Type	Description
Physical	Single slot, double height VME Eurocard form factor board: Dimensions: 233 mm by 160 mm
Electrical	Voltage: 4.75 Vdc to 5.25 Vdc Current: 4.0 Amps typical (at +5 Vdc)
Capacity	Up to fifteen SCSI Devices
Transfer Rate	SCSI data rate to 7 Mbytes per second in Asynchronous mode. SCSI data rate up to 20 Mbytes per second in Synchronous mode.
Environmental Operating	Temperature: 0° to +55°C Air Flow: 200 linear feet per minute Humidity 10% to 80% (non-condensing) Elevation: 0 feet to 10,000 feet
Non-Operating	Temperature: -40° to + 85°C Humidity: 10% to 95% (non-condensing) Elevation: 40,000 feet maximum
Bus Interface	VMEbus Standard (Revision D)
Drive Interface	Draft Proposed American National Standard: Small Computer Systems Interface-2, (ANSI X3.131 - 199x Draft Proposed American National Standard for information systems- SCSI-3 Parallel Interface)

Table 124: Specifications for the Rf3880

D

Defaults

Introduction

Each Rimfire 3880 is shipped with the same factory settings. If you change nothing when you receive your board, it will power-up with the defaults that are listed in this appendix.

Summary

This appendix provides information on the following:

- Hardware defaults as shipped from the factory.
- Firmware defined defaults for these Board-control commands: Identify, General Options, Unit Options and Extended Unit Options, and Board Information.

This appendix is helpful when you wish to know the values the RF3880 will assume if you do not change an option.

Hardware Defaults

This section describes the defaults you will find for both the jumpers on the board, and the Hardware Ports you will use to set-up the board and operate it.

Board Jumpers

These are the factory settings for the jumpers on the board. For more information see Chapter 3.

Jumper	Default Setting	Meaning
VMEbus Address Modifier	No jumper at AM2	2D, Short Supervisory Access
VMEbus Address	Jumper at A12	EE00H
SCSI Configuration Jumper Block:		
Parity Checking	OUT	Enabled
SCSI Bus Reset at Power-up	OUT	Enabled
RF3880 SCSI ID	OUT	ID = 0
Bus Request/Grant Level	6 jumpers installed: pins 4-5 under 0 pins 4-5 under 1 pins 4-5 under 2 pins 1-2, 3-4, 5-6 under 3	Level 3
Sysfail	IN	Sysfail signal asserted on bus during Reset.
SCSI Bus Terminators	IN	RF3880 provides SCSI Bus Termination.

Table 125: Hardware Defaults - Jumpers

Hardware Ports

These are the values that will be written if you do not specify other values. See Chapter 2 for more information.

Port	Default	Meaning
Address Buffer Port:		
Byte Swapping Control	0	No Swapping
Word Swapping Control;	0	No Swapping
Width	1	32 bit or 64 bit transfers, dependent on address modifier used.
Status Port:		
Entered Bit	0	Initial value.
Ready Bit	1	After Reset has completed.
Code	02H	Whenever ERR bit is zero, this should be 02H.
Channel Attention Port	There is no default: Write a 2 to acknowledge receipt of Target Selection Data Structure. Write a 1 to initiate a Command List. Write a 0 to issue a Single Command	
Reset Port	There is no default: Write any value to this port to get Reset.	

Table 126: Defaults for Hardware Ports

Command Defaults

If you do nothing to change the board operation by using the available Board-control commands, the adapter uses defaults. These are listed below. For more information see Chapter 7.

Command Parameters		Default Values	Meaning
Identify - returned in the status block			
Engineering Revision		XX	Depends on revision.
Firmware Revision		XX	
Day, Month, Year		XXXXXX	
Options Flags = 14H			
	Tag	bit 2 = 1	Tagged commands supported.
	Reset	bit 4 = 1	SCSI bus will reset at adapter reset - set by board jumper default setting.
	SCSI ID	bits 0, 5, 6, 7 = 0	Set by board jumper default settings.
FW # of Even Prom		XXXXXXXX	Depends on revision.
General Option - these are used if you set nothing yourself			
Bus Throttle		0BH	2048 bytes per burst.
Select Flags = 03H			
	Disconnect/Reconnect	bit 0 = 1	Allow Disconnect/Reconnect.
	SCS Bus Parity	bit 1 = 1	Check for SCSI parity errors.
	Block Mode Transfers	bit 2 = 0	Block Mode transfers depend on address modifiers.
Unit Options and Extended Unit Options - used if you set nothing			
Disconnect Timeout		0H	No timeout occurs
Retry Limit		0H	No retries enabled.
Retry Control = 0H			
	Issue Interrupt	bit 0 = 0	No interrupt on retry.
	Issue Status Block	bit 1 = 0	No Status Block on retry.
	Retry Parity Errors	bit 2 = 0	No retry for parity errors.
	Retry Command Errors	bit 3 = 0	No retry for device errors.
	Retry Bus Errors	bit 4 = 0	No retry for bus errors.
Select Timeout		FAH	250 milliseconds.
Unit Flags = 0H			
	Untagged Queue	bit 0 = 0	Send one command at a time.
	Synch Negotiation	bit 1 = 0	Will not initiate synchronous negotiation.
	Inhibit ATN Signal	bit 2 = 0	ATN will be asserted.
	Wide Transfers	bit 3 = 0	Will not initiate wide negotiation.
	Tagged Commands	bit 6 = 0	Tagged commands not negotiated.
Sense Count		08H	First eight bytes of Sense Data.
Selected Sense Bytes		all = 0	First eight bytes of Sense Data returned.
Board Information - these are returned in a Data Structure			
Select Flags = 03H			
	Disconnect/Reconnect	bit 0 = 1	Allow Disconnect/Reconnect.
	SCSI Bus Parity	bit 1 = 1	Check for SCSI parity errors.
	Block Mode Transfers	bit 2 = 0	Block Mode transfers depend on address modifiers.
Bus Throttle		0BH	2048 bytes per burst.

Table 127: Command Defaults

Command Parameters	Default Values	Meaning
Engineering Revision	XX	Depends on revision.
Firmware Revision	XX	
Day, Month, Year	XXXXXX	
Options Flags = 14H		
TAG	bit 2 = 1	Tagged commands supported.
Reset	bit 4 = 1	SCSI bus will reset at adapter reset - set by board jumper default setting.
SCSI ID	bits 0, 5, 6, 7 = 0	Set by board default jumper.
FW# of Even Prom	XXXXXXXX	Depends of revision.
Termination = 0H		
Bad SCSI Term Power	bit 1 = 0	TERM PWR is good.
Blown SCSI Fuse	bit 2 = 0	Fuse us good.
Jumper Configuration	0H	All jumpers removed.
Adapter Base Model	3880H	RF3880 family of adapters.
Target - Disconnect Timeout	0H	No timeout occurs.
Target - Retry Limit	0H	No retries occur.
Target - Sense Count	08H	First eight bytes of Sense Data.
Target - Device Flags = 0		
Untagged Queueing	bit 0 = 0	Allow 1 Command at a time.
Synch Negotiation	bit 1 = 0	Synchronous negotiation no initiated
Inhibit ATN Signal	bit 2 = 0	ATN will be asserted.
Wide Negotiation.	bit 3 = 0	Wide negotiation not initiated.
Tagged Queueing	bit 6 = 0	Tagged Queueing NOT enabled.
Issue Interrupt	bit 8 = 0	No interrupt for retry.
Issue Status Block	bit 9 = 0	No Status Block per retry.
Retry Parity Errors	bit 10 = 0	No retry for Parity Errors.
Retry Command Errors	bit 11 = 0	No retry for command errors.
Retry Bus Errors	bit 12 = 0	No retry for bus errors.
Selected Sense	bit 14 = 0	First 8 bytes of sense data returned.
Target - Synch Period	† 19H	Equates to 100 nanoseconds.
Target - Synch Offset	†† 0FH	QLogic chip limit. (15 REQ/ACK’s)
Target - Width	00H	8 bits
Target - Selected Sense Bytes	all = 0	First eight bytes of Sense Data returned.

Table 127: Command Defaults

Note
† This value is returned when Synchronous transfer rate is 10 MHz.
†† This value is returned if the full QLogic FIFO is in use.
Both values are zero if operation is in Asynchronous mode.

E

Design Differences

Introduction

Although the RF3880 is compatible with previous Ciprico VME/SCSI adapters, it is designed to new industry specifications and therefore has some differences that you may need to note if you are upgrading a driver that was written for one of Ciprico's previous products.

Summary

- Changes in supported features
- Changes in implementation of features

This appendix is useful to anyone who is modifying an existing driver for use with the RF3880.

To make differences easy to find, the following descriptions include page numbers that reference other parts of this User's Guide.

Features no longer Supported

The following design features are not supported in the RF3880:

Scatter/Gather Operations

The RF3880 does not support the Scatter/Gather feature that was available in previous adapters. If your driver used the Scatter/Gather feature, you must now issue individual SCSI commands for each descriptor pair in the Scatter/Gather Descriptor Block.

Byte and Word Swapping of Data

The RF3880 does not support word swapping and byte swapping of data. Thus, Bits 3 and 4 of the Control Field of the Address Buffer Port must be zero. This is shown in Table 6 on page 2 - 8. Any required data swapping must now be done by the driver/system. Note that the RF3880 still supports swapping of Control Structures.

Odd Byte Handling

The RF3880 does not support a selectable Odd Byte Handling algorithm. In previous designs, Odd Byte Handling was selected in the General Options Command, Bit 3 of the Select Flags field. As shown in Table 73 on page 7 - 16, Bit 3 of this field is now ignored.

Previously, the Odd Byte Handling algorithm currently selected was reported in the Board Information Command, Bit 3 of the Select Flags Field. As shown in Table 92 on page 7 - 34, Bit 3 is now returned as a zero.

Implementation Differences

The following descriptions explain changes made to the way the RF3880 implements features:

Width of Data Transfers on the VMEbus

Because the RF3880 implements new bus specifications for VME D64, there have been changes to the way that the VMEbus width is selected.

The width of any data transfer now depends on the Address Modifier supplied, and the value of the WID Bit in the Control Field of the Address Buffer Port. The WID bit is shown in Table 6 on page 2 - 8. When WID is 0, 16-bit transfers are forced. When WID is 1, 32 or 64 bit transfers will be used, dependent on the Address Modifier used. Address Modifiers 08H, 0CH, 38H, and 3CH will cause 64 bit transfers.

Note that the width of data transfers also depends naturally on the actual data transfer address. Complete 16, 32, or 64 bit transfers can only occur when the address has the proper alignment (to word, longword, or double-longword). The RF3880 will perform smaller width transfers until the proper address alignment is reached. The adapter will then shift into the maximum width allowed by the WID bit and the Address Modifier.

Status Port Error Codes and Test Flags

The RF3880 has a new, high performance hardware architecture. This new hardware requires different Status Port Error Codes. These are shown in Table 10 on page 2 - 13. Status Port Error Codes 14H, 1CH, and codes C0H to CCH are no longer used. Codes 80H, 84H, 88H, 8CH, and D0H were added.

Additionally, in the Diagnostic Command’s Test Flag field, Bit 3 is now the RAM test bit. See Table 79 on page 7 - 24.

Extended SCSI ID’s

The SCSI-3 specification, in addition to allowing 16-bit data transfers, supports 16 SCSI ID’s. (Previous products were limited to eight.) Changes were made in the following areas to accommodate the additional SCSI ID’s:

SCSI Configuration Jumper Block - A hardware jumper for the extra bit (SCSI ID Bit 3) of the SCSI ID was necessary. Changes were made to the SCSI Configuration Jumper Block in order to implement this. The new arrangement of the jumper block is shown in Table 13 on page 3 - 6.

The Board Information Command reports the jumper setting of the SCSI Configuration Jumper Block, in the SCSI Cfig Block byte. Since the jumpers were rearranged, the value expected in this byte might change. See Table 98 on page 7 - 37.

Identify and Board Information Commands - Both of these commands return an Option Flags field that now uses Bit 0 to report SCSI ID Bit 3. For Identify command see Table 64 on page 7 - 10. For the Board Information command, see Table 92 on page 7 - 34.

Additionally, the Board Information Command Parameter Block has a new field, shown on Table 91 on page 7 - 33. Called the Options field, its purpose is to allow you choose whether information for all possible SCSI ID’s is returned or information is limited to eight.

Unit Options and Extended Unit Options Commands - The Unit SCSI ID byte in both commands can now contain a number from 0 to 0FH.

Board Statistics Command - The Statistics Data Structure returned from a Board Statistics command now reports statistics for units at SCSI ID's 8 through 15 starting at Offset 28H. This is shown in Table 68 on page 7 - 12. These bytes were reserved on the RF3560/70 and RF3870. The RF3510 used these bytes for Floppy Statistics.

Extended Board Statistics Command - The Extended Board Statistics Command Parameter Block has a new field, shown in Table 100 on page 7 - 41. Called the Options field, its purpose is to allow you choose whether statistics for all possible SCSI ID's is returned in the Extended Board Statistics Data Structure (length D4H bytes) or is limited to eight (length 74H bytes).

Bus Throttle

The RF3880 uses a new scheme to control the number of VME transfers to make each burst. The Bus Throttle byte specifies the number of bytes in a burst as a power of 2. Valid values are 0 to 0BH. Any values larger than 0BH will be changed to 0BH.

The value of the Bus Throttle byte is set in the General Options Command and is reported in the Board Information Command Data Structure.

It is valuable to note that VME D64 Block Modes limit the size of a burst on the bus to 2048 bytes per burst. To take advantage of this, the data transfer addresses you use must be aligned on 2048 byte boundaries. Any other alignment and the burst size will be limited to 256 bytes per burst.

Non VME D64 Block Modes limit the burst size on the bus to 256 bytes per burst.

Non Block Modes allow burst sizes to 2048 bytes per burst.

Block Mode Transfers

The RF3880 assigns a new meaning to the BMT bit that is part of the Select Flags field of the General Options Command.

SCSI Data Widths

The RF3880 supports SCSI data widths of 8 or 16 bits. Negotiation for 16 bit wide SCSI is enabled by setting Bit 3 (WID) of the Unit Options Command's Unit Flags field, shown in Table 76 on page 7 - 20 or by setting the same bit in the Unit Flags field of the Extended Unit Options command, shown in Table 106 on page 7 - 48.

Additionally, the Board Information Command reports the current setting of the bit in it's Device Flags Field and the Per Target-Width field reports the result of the negotiation.

Uniquely Identify the RF3880

In order to differentiate the RF3880 from the RF3560/70 and RF3870, a new word field was added to the Board Information Data Structure returned from the Board Information command. This new word is the Adapter Base Model # (number) word.

Odd-byte Addressing

VME addresses for transfers must be word-aligned.

How to Maximize Performance

In order to take full advantage of the RF3880 design, you should do the following:

- Align VME D64 data buffers to 2K address boundaries, and specify a Bus Throttle of 2048 bytes per burst. Any other alignment will reduce the maximum Burst size to 256 bytes per burst.
- Align all data buffers to double-longword address boundaries. Any other alignment will cause multiple data transfer cycles on the VME bus. The first transfer will move enough data to align the address to a double-longword. The next transfer will move the data according to the Bus Throttle setting.

Index